

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
МІСЬКОГО ГОСПОДАРСТВА ІМЕНІ О. М. БЕКЕТОВА  
Навчально-науковий інститут енергетичної, інформаційної  
та транспортної інфраструктури

Кафедра комп'ютерних наук та інформаційних технологій

Пояснювальна записка  
до кваліфікаційної роботи бакалавра

на тему: «Проект інформаційної вебплатформи для обліку продукції та онлайн-замовлень агропромислового підприємства»

Виконав: студент 4 курсу, групи КН 2022-1  
спеціальності

122 Комп'ютерні науки

(шифр і назва спеціальності)

Максим КОРНІЄНКО

(ім'я та прізвище)

Керівник: Юрій ПАХОМОВ

(ім'я та прізвище)

Рецензент Микола КАРПЕНКО

(ім'я та прізвище)

м. Харків – 2026 рік

Харківський національний університет міського господарства імені О. М. Бекетова

(повне найменування закладу вищої освіти)

Навчально-науковий Інститут енергетичної, інформаційної

та транспортної інфраструктури

Кафедра комп'ютерних наук та інформаційних технологій

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 Комп'ютерні науки

(шифр і назва)

**ЗАТВЕРДЖУЮ**

Завідувач кафедри КНтаІТ



Марина НОВОЖИЛОВА

« 23 »      червня      2026 року

**З А В Д А Н Н Я**  
**НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Корнієнку Максиму Вячеславовичу

(прізвище, ім'я, по батькові)

1. Тема роботи «Проект інформаційної вебплатформи для обліку продукції та онлайн-замовлень агропромислового підприємства»

керівник роботи Пахомов Ю.В., к.т.н., доцент кафедри КНтаІТ

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом закладу вищої освіти від 22.05.2026; наказ №440-03

2. Термін подання студентом роботи 17.06.26









3. Вихідні дані до роботи Аналіз бізнес-процесів агропромислового підприємства з виробництва комбікормів, HTML5, CSS3, JavaScript, Web Storage API (localStorage), CSV.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): Аналіз предметної області агропромислового виробництва, виявлення ключових проблем десинхронізації даних, ручної обробки замовлень. Визначаються функціональні вимоги до вебплатформи, інтерактивного каталогу з фільтрацією продукції, реалізація механізму онлайн-замовлень, особистих кабінетів та спеціалізованої адміністративної панелі з аналітикою. Розглядається структура даних на основі Web Storage API із використанням JSON. Описується програмна реалізація інтерфейсу за архітектурою Single Page Application із застосуванням HTML5, CSS3 та JavaScript, механізми експорту даних у формат CSV, а також наводиться керівництво користувача.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):

презентація

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Юрій ПАХОМОВ, к.т.н., доцент каф. КНтаІТ	26.05.26 	02.06.26 
2	Юрій ПАХОМОВ, к.т.н., доцент каф. КНтаІТ	02.06.26 	09.06.26 
3	Юрій ПАХОМОВ, к.т.н., доцент каф. КНтаІТ	05.06.26 	13.06.26 
4	Вікторія МАЛИШЕВА, к.т.н., доцент кафедри безпеки життєдіяльності	06.06.26 	16.06.26 

7. Дата видачі завдання 25.05.26**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів кваліфікаційної роботи	Строк викон. етапів роботи	Примітка
1	Вибір теми кваліфікаційної роботи	26.05.2026	Викон.
2	Затвердження тем, наукових керівників, завдань та календарного плану підготовки кваліфікаційної роботи	26.05.2026	Викон.
3	Написання I розділу	02.06.2026	Викон.
4	Написання II розділу	09.06.2026	Викон.
5	Написання III розділу	13.06.2026	Викон.
6	Написання IV розділу Охорона праці	16.06.2026	Викон.
8	Подання кваліфікаційної роботи керівнику	17.06.2026	Викон.
9	Робота по усуненню зауважень керівника, уточнення і доповнення практичного матеріалу, оформлення додатків до роботи	18.06.2026	Викон.
11	Захист матеріалів кваліфікаційної роботи на засіданні кафедри	20.06.2026	Викон.
12	Офіційний захист матеріалів кваліфікаційної роботи на засіданні екзаменаційної комісії	25.06.2026	Викон.

Студент

(підпис)

Корнієнко М. В.

(прізвище та ініціали)

Керівник роботи

(підпис)

Пахомов Ю. В.

(прізвище та ініціали)

## АНОТАЦІЯ

Дана кваліфікаційна робота присвячена проектуванню та розробці інформаційної вебплатформи для агропромислового підприємства, що спеціалізується на виробництві комбікормів та кормових добавок. Проведено системний аналіз предметної області, який виявив проблеми розсинхронізації даних, впливу людського фактора під час ручної обробки замовлень та обмеженості обслуговування контрагентів у часі.

Розроблена система пропонує користувачам інтерактивний каталог із модулем багатофакторної фільтрації продукції за призначенням для різних видів сільськогосподарських тварин. Клієнтський інтерфейс забезпечує можливість перевірки актуальних складських залишків у режимі реального часу, формування онлайн-замовлень через електронний кошик та ведення прямої комунікації з адміністрацією через систему тікетів у персональному кабінеті.

Для персоналу підприємства реалізовано закриту адміністративну панель. Модуль надає інструментарій для диспетчеризації замовлень, управління клієнтською базою та контролю критичних залишків номенклатури. Інтегрований аналітичний дашборд дозволяє візуалізувати купівельний попит за допомогою векторних динамічних графіків.

Програмну реалізацію виконано за архітектурою Single Page Application (SPA) з використанням базових вебтехнологій HTML5, CSS3 та JavaScript. Збереження даних функціонує на базі Web Storage API (localStorage) зі структурованими JSON-колекціями, що гарантує автономність роботи прототипу. Для інтеграції вебплатформи з бухгалтерськими програмами підприємства впроваджено механізм автоматичного експорту масивів даних у формат CSV.

Ключові слова: вебплатформа, агропромислове підприємство, електронна комерція, Single Page Application, облік продукції, JavaScript.

## ANNOTATION

This qualification work is devoted to the design and development of an information web platform for an agro-industrial enterprise specializing in the production of animal feed and feed additives. A systematic analysis of the subject area was conducted, which revealed the problems of data desynchronization, the influence of the human factor during manual order processing, and time constraints in servicing counterparties.

The developed system offers users an interactive catalog with a multi-factor product filtering module based on its intended use for various farm animals. The client interface provides the ability to check up-to-date warehouse balances in real time, place online orders via an electronic shopping cart, and conduct direct communication with the administration through a ticket system in the personal account.

A closed administrative panel has been implemented for the enterprise staff. The module provides a set of tools for order dispatching, customer base management, and control over critical inventory shortages. An integrated analytical dashboard allows visualizing purchasing demand using dynamic vector charts.

The software implementation was executed using the Single Page Application (SPA) architecture with basic web technologies: HTML5, CSS3, and JavaScript. Data storage operates on the basis of the Web Storage API (localStorage) with structured JSON collections, ensuring the autonomous operation of the prototype. To integrate the web platform with the enterprise's accounting software, a mechanism for the automatic export of data arrays into CSV format has been introduced.

Keywords: web platform, agro-industrial enterprise, e-commerce, Single Page Application, product accounting, JavaScript.

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	7
ВСТУП .....	8
РОЗДІЛ 1 ЗАГАЛЬНІ ПОЛОЖЕННЯ.....	10
1.1 Опис предметного середовища.....	10
1.2 Огляд наявних аналогів .....	15
1.3 Постановка задачі .....	20
1.4 Висновки до розділу 1 .....	21
РОЗДІЛ 2 ІНФОРМАЦІЙНЕ ТА МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ .....	23
2.1 Аналіз предметної області .....	23
2.2 Моделювання бізнес-процесів агропромислового підприємства .....	26
2.3 Висновки до розділу 2 .....	30
РОЗДІЛ 3 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ.....	33
3.1 Проектування та реалізація структури збереження даних .....	33
3.2 Програмна реалізація.....	44
3.3 Керівництво користувача .....	65
3.4 Висновки до розділу 3 .....	71
РОЗДІЛ 4 ОХОРОНА ПРАЦІ.....	73
4.1 Організаційно-правові основи забезпечення безпеки праці.....	73
4.2 Характеристика об'єкта та виявлення потенційних небезпек .....	75
4.3 Дослідження ризику реалізації потенційних небезпек на об'єкті проектування та розробка заходів щодо їх попередження.....	78
4.4 Висновки до розділу 4 .....	82
ВИСНОВКИ.....	84
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	86

## ПЕРЕЛІК СКОРОЧЕНЬ

API – Application Programming Interface (інтерфейс програмування додатків).

B2B – Business-to-Business (бізнес для бізнесу, комерційні відносини між компаніями).

B2C – Business-to-Consumer (бізнес для споживача, комерційні відносини з кінцевими клієнтами).

BOM – Byte Order Mark (мітка порядку байтів).

CMS – Content Management System (система управління контентом).

CRM – Customer Relationship Management (система управління відносинами з клієнтами).

CSS – Cascading Style Sheets (каскадні таблиці стилів).

CSV – Comma-Separated Values (текстовий формат обміну даними, значення, розділені комою).

DOM – Document Object Model (об'єктна модель документа).

ERP – Enterprise Resource Planning (планування ресурсів підприємства).

HTML – HyperText Markup Language (мова розмітки гіпертексту).

JS – JavaScript (мова програмування).

JSON – JavaScript Object Notation (текстовий формат обміну даними).

KPI – Key Performance Indicator (ключовий показник ефективності).

MIME – Multipurpose Internet Mail Extensions (багатоцільові розширення пошти Інтернету, стандарт ідентифікації типів файлів).

SEO – Search Engine Optimization (пошукова оптимізація).

SPA – Single Page Application (односторінковий вебзастосунок).

SVG – Scalable Vector Graphics (масштабована векторна графіка).

UI – User Interface (користувацький інтерфейс).

UX – User Experience (користувацький досвід).

WMS – Warehouse Management System (система управління складом).

## ВСТУП

Актуальність теми. В умовах глобальної цифровізації економіки агропромисловий комплекс України потребує глибокої модернізації бізнес-процесів. Традиційні підходи, що базуються на паперовому документообігу та ручному управлінні замовленнями, втрачають свою ефективність і стають бар'єром для розвитку, особливо у сфері виробництва комбікормів. У сучасних умовах для агропідприємств стає критично важливим впровадження спеціалізованих вебплатформ, які забезпечують автоматизацію рутинних операцій, мінімізують вплив людського фактору, оптимізують логістичні ланцюжки та підвищують рівень клієнтського сервісу.

Мета та завдання роботи. Метою даної роботи є проектування та розробка інформаційної вебплатформи для агропромислового підприємства (виробника комбікормів), яка забезпечить комплексну автоматизацію процесів складського обліку готової продукції та зручне оформлення онлайн-замовлень користувачами. Для досягнення поставленої мети необхідно вирішити такі завдання: провести комплексний аналіз предметної області та бізнес-процесів підприємства; здійснити порівняльний аналіз існуючих на ринку рішень; сформулювати функціональні та нефункціональні вимоги до системи; спроектувати структуру бази даних; розробити адаптивний інтерфейс користувача; реалізувати програмну логіку вебплатформи; провести тестування розробленої системи.

Об'єкт та предмет дослідження. Об'єктом дослідження є інформаційна вебплатформа (вебзастосунок) для агропромислового підприємства з виробництва комбікормів як інструмент взаємодії користувачів із системою та автоматизації складського обліку. Предметом дослідження виступають методи та технології розробки сучасних клієнтських вебзастосунків (за архітектурою Single Page Application) із використанням HTML, CSS, мови

програмування JavaScript, а також механізмів локального збереження даних та генерації звітності.

Методи дослідження. Для виконання завдань використано методи системного аналізу для дослідження предметної області, методи порівняльного аналізу для оцінки конкурентних програмних продуктів, принципи проєктування реляційних баз даних та користувацьких інтерфейсів (UI/UX), а також методи тестування програмного забезпечення для перевірки його працездатності та стабільності.

Наукова та практична новизна. Наукова новизна роботи полягає у проєктуванні архітектури системи, що органічно об'єднує функції фронтенд-рішень електронної комерції (e-commerce) з бекенд-інструментами управління складом (WMS/ERP-елементи). Практична значущість полягає у створенні готового до впровадження програмного продукту, використання якого дозволить реальному підприємству скоротити час на обробку клієнтських заявок, знизити кількість операційних помилок при обліку товарів та відкрити новий ефективний канал збуту продукції через мережу Інтернет.

Структура роботи. Робота складається зі вступу, трьох основних розділів, висновків та списку використаних джерел. У першому розділі проаналізовано специфіку бізнес-процесів підприємства з виробництва комбікормів, виявлено проблеми обліку продукції та проведено огляд рішень конкурентів. Другий розділ присвячено дослідженню інформаційного середовища та моделюванню бізнес-процесів. У третьому розділі викладено технічні вимоги, проєкт архітектури даних та описано безпосередню програмну реалізацію вебплатформи. Висновки містять підсумки виконаної роботи та результати впровадження системи.

## РОЗДІЛ 1 ЗАГАЛЬНІ ПОЛОЖЕННЯ

### 1.1 Опис предметного середовища

Виробництво комбінованих кормів (комбікормів) є однією з найбільш складних, багатокomпонентних та динамічних галузей сучасного агропромислового комплексу (АПК). Сьогодні підприємство цього профілю функціонує не просто як традиційний млин для переробки зерна, а як високотехнологічний виробничий комплекс. Його головна мета – створення збалансованих, науково обґрунтованих раціонів для різних видів сільськогосподарських тварин (птиці, свиней, великої рогатої худоби, риби тощо) з чітким урахуванням їхніх вікових груп, фізіологічного стану та виробничих потреб.

Специфіка галузі зумовлює управління величезною номенклатурою продукції, яка може налічувати сотні активних позицій. До основних категорій належать:

1. Повнораціонні комбікорми (ПК) – повністю забезпечують потребу тварин у поживних речовинах.
2. Білково-мінерально-вітамінні добавки (БМВД) – концентровані суміші для збагачення основного раціону.
3. Премікси – мікродобавки біологічно активних речовин.
4. Замінники незбираного молока (ЗНМ) та спеціалізовані лікувальні або стартові корми.

Управління таким широким асортиментом, що постійно оновлюється залежно від наявної сировини та сезону, вимагає бездоганного складського та виробничого обліку.

Особливості клієнтської бази та каналів збуту Клієнтська база виробника комбікормів є вкрай неоднорідною, що вимагає застосування різних підходів до обслуговування.

З одного боку, підприємство працює у B2B сегменті (великі фермерські господарства, птахофабрики, агрохолдинги). Ці клієнти замовляють продукцію великими оптовими партіями (десятками тон), укладають довгострокові контракти, мають специфічні вимоги до логістики (доставка кормовозами) та індивідуальні багаторівневі системи знижок. Часто для таких замовників виготовляються корми за індивідуальною рецептурою, адаптованою під конкретні лабораторні аналізи їхньої власної сировини.

З іншого боку, існує великий B2C та дрібний B2B сегмент (приватні домогосподарства, локальні фермери). Вони купують корми у фасованому вигляді (мішки по 10–50 кг) для власних потреб. Для цієї категорії споживачів критично важливими є наявність зручного каталогу, детальний опис складу продукту, прозорість роздрібних цін та можливість швидкого оформлення замовлення з доставкою поштовими операторами або самовивозом.

Проблематика традиційних методів управління За відсутності єдиної інтегрованої цифрової платформи, бізнес-процеси підприємства стикаються з низкою критичних обмежень:

1. Ручна обробка замовлень: Великий штат менеджерів з продажу змушений працювати в ручному режимі – приймати заявки через телефонні дзвінки, розрізнені месенджери (Viber, Telegram) або електронну пошту.

2. Десинхронізація даних: Менеджери постійно витрачають час на звірку залишків продукції на складах, перемикаючись між програмами внутрішнього бухгалтерського обліку та локальними електронними таблицями (Excel). Це часто призводить до ситуацій, коли один і той самий товар може бути випадково обіцяний двом різним клієнтам.

3. Високий вплив «людського фактора»: При ручному перенесенні даних із повідомлень клієнтів у систему обліку виникають помилки в номенклатурі, об'ємах або адресах доставки, що призводить до фінансових та репутаційних втрат.

4. Низька пропускна здатність: Цей процес є повільним та непрозорим для клієнта. Він створює жорсткі «вузькі місця» в періоди сезонних піків

попиту (наприклад, весняний період вигодовування молодняку), коли менеджери фізично не встигають обробити всі вхідні звернення.

Враховуючи вищезазначені фактори, цифровізація бізнес-процесів через створення власної інформаційної вебплатформи є не просто модним трендом, а необхідним еволюційним кроком для виживання та масштабування бізнесу [1].

Впровадження системи, яка поєднує інструменти електронної комерції з автоматизованим обліком продукції, дозволить перевести клієнтів на формат самообслуговування (Self-Service) [2]. Користувачі зможуть 24/7 переглядати актуальний каталог, бачити реальні залишки на складах, самостійно формувати замовлення в особистому кабінеті та відстежувати їхній статус. Для підприємства це означає кардинальне зниження операційного навантаження на персонал, мінімізацію помилок, прискорення циклу виконання замовлень та забезпечення абсолютної прозорості торговельних операцій на всіх етапах.

На етапі проектування будь-якої спеціалізованої інформаційної платформи критично важливим кроком є проведення глибокого системного аналізу існуючої проблематики, з якою щоденно стикається агропідприємство. Дослідження типових операційних, логістичних та комунікаційних бізнес-процесів виявило низку фундаментальних проблем, які гальмують розвиток компанії та потребують негайного програмного вирішення шляхом комплексної автоматизації.

Існує декілька проблемних зон на сьогодні, проблема розсинхронізації даних та відсутність «єдиного джерела правди» (Data Silos) – є найбільш поширеною та критичною проблемою є фрагментація інформаційного простору підприємства. Інформація про фактичну наявність продукції зберігається у закритій складській системі обліку (наприклад, 1С або ізольованих таблицях Excel). Водночас каталог продукції для клієнтів існує у вигляді друкованих прайс-листів або статичних PDF-документів на застарілому корпоративному сайті, а дані про поточні замовлення фіксуються

у відокремленій CRM-системі або взагалі в паперових блокнотах менеджерів. Внаслідок такої розсинхронізації виникає інформаційний вакуум: коли клієнт планує зробити замовлення, він не бачить реальних складських залишків. Це часто призводить до ситуацій, коли менеджер приймає та підтверджує замовлення на позицію, яка щойно була відвантажена іншому контрагенту. Як наслідок – зрив строків постачання, необхідність оформлення повернення коштів, екстрені й економічно не вигідні зміни у виробничому плані та суттєві репутаційні втрати для підприємства.

Другою проблемою є критичний вплив «людського фактора» та висока ціна операційних помилок. При ручному перенесенні даних про замовлення (наприклад, продиктованих клієнтом по телефону або надісланих у месенджері) в основну систему обліку підприємства виникає явище «зіпсованого телефону». Специфічна номенклатура комбікормів вимагає максимальної точності. Наприклад, помилково оформлене замовлення на корм для курчат-бройлерів (переплутані рецептури «Старт», «Гровер» або «Фінішер») може порушити весь цикл вигодовування на фермі замовника, що призведе до захворюваності тварин та величезних фінансових претензій до виробника. Окрім помилок в артикулах, ручне введення часто супроводжується помилками у вагових об'ємах або адресах доставки. Відправка багатотонної вантажівки за хибним маршрутом призводить до колосальних неочікуваних логістичних збитків.

Обмеженість у часі та відсутність доступу до сервісу 24/7 також є одною з основних проблем. Традиційний формат роботи відділу збуту жорстко обмежується стандартними робочими годинами менеджерів (переважно з 9:00 до 18:00). Проте специфіка аграрного сектору полягає в тому, що фермери та керівники господарств працюють у ненормованому графіку, часто з раннього ранку до пізньої ночі. Вони позбавлені можливості отримати консультацію або сформулювати замовлення у зручний для них час. Сучасний електронний бізнес (e-commerce) диктує нові стандарти обслуговування: клієнт повинен мати безперебійний доступ до каталогу та можливість здійснити покупку в будь-

який момент часу (24/7), використовуючи будь-який зручний пристрій (насамперед смартфон, перебуваючи безпосередньо на виробничому об'єкті).

Агропідприємства, що виробляють комбікорми, використовують надзвичайно складну, багатофакторну матрицю ціноутворення. Кінцева ціна для контрагента формується динамічно і залежить від:

1. Обсягу партії (роздрібний мішок, палета, тонна, вагонна норма);
2. Історії співпраці (накопичувальні знижки для постійних партнерів);
3. Регіону та умов доставки (самовивіз або логістика виробника);
4. Умов оплати (повна передоплата, часткова оплата або відтермінування платежу за договором).

У ручному режимі розрахунок вартості такого замовлення забирає багато часу та вимагає численних погоджень з керівництвом. Вебплатформа має вирішити цю проблему шляхом впровадження системи ролей: після авторизації система повинна автоматично ідентифікувати клієнта, аналізувати його статус і миттєво відображати персоналізовані ціни у каталозі.

Ефективне управління неможливе без опори на актуальні дані (Data-Driven Decision Making). За існуючої моделі керівництво підприємства не має можливості оперативно, в розрізі одного дня або тижня, відстежувати динаміку продажів за окремими товарними категоріями, регіонами присутності або конкретними клієнтами. Звіти зазвичай формуються бухгалтерією постфактум – наприкінці місяця або кварталу. Це унеможливорює швидке управлінське реагування на ринкові зміни: підприємство не може вчасно помітити просідання попиту на певні види сезонних кормів, попередити переповнення складів нереалізованою продукцією або вчасно закупити необхідний обсяг дефіцитної сировини.

Аналіз описаних вище проблем доводить, що їх усунення неможливе шляхом простого розширення штату працівників. Єдиним ефективним рішенням є проектування та впровадження комплексної інформаційної вебплатформи, яка об'єднає розрізнені бази даних в єдину екосистему,

автоматизує рутинні процеси, виключить «людський фактор» та забезпечить безперервну взаємодію між виробником та кінцевим споживачем.

## 1.2 Огляд наявних аналогів

Для формування обґрунтованих вимог до архітектури та функціоналу майбутньої інформаційної платформи, а також для створення конкурентоспроможного програмного продукту, було проведено бенчмаркінг (порівняльний аналіз) існуючих рішень на українському ринку в ніші реалізації комбікормів та агрохімії.

Аналіз ринку демонструє високий ступінь фрагментації цифрових рішень. Більшість існуючих сайтів не задовольняють комплексні потреби сучасного агробізнесу, оскільки вони або виконують виключно інформаційну функцію, або оптимізовані лише під дрібний роздріб [3]. Для детального розгляду обрано два реальні діючі веб-ресурси, що репрезентують кардинально протилежні підходи до ведення бізнесу в мережі Інтернет: корпоративний сайт великого постачальника «Компанія Агрокапітал України» та шаблонний роздрібний інтернет-магазин «Кормобаза».

Першим проведемо аналіз вебплатформи "Компанія Агрокапітал України".

The screenshot shows the website 'АГРОКАПІТАЛ УКРАЇНИ'. The header includes the company logo, contact information (+38 (044) 580-0-580), and a navigation menu with items: КАТАЛОГ ПРОДУКЦІЇ, ЦІНИ, НОВИНИ, КОНТАКТИ, ІНТЕРНЕТ-МАГАЗИН. The main content area is titled 'Комбікорми від виробника' and contains text about the company's products, contact details for wholesale, and a list of advantages of using combined feed. A sidebar on the left lists various product categories like 'ПРОДУКЦІЯ ДЛЯ ХАРЧОВОЇ ПРОМИСЛОВОСТІ', 'КОМБІКОРМИ', 'КОРМОВІ ДОБАВКИ', etc. A small pop-up window at the bottom right asks for consent to use cookies.

## Рисунок 1.1 – Вебзастосунок «agrokapital.com.ua»

Цей ресурс є класичним прикладом типового корпоративного сайту-візитки великого постачальника, який, маючи представництво в інтернеті, фактично продовжує працювати виключно за традиційною офлайн-моделлю "телефонних продажів". Інтерфейс виконує роль вітрини, але не забезпечує замкненого циклу обслуговування.

Виявлено чотири недоліки та архітектурні проблеми порівняно з розроблюваним рішенням.

Першим є відсутність повноцінного онлайн-замовлення та кошика. На сайті розміщено статичний каталог продукції, але повністю відсутня функція додавання товару в кошик. Оформлення електронного замовлення через інтерфейс платформи є неможливим. Це критично обриває конверсійну воронку: клієнт, який готовий здійснити покупку ввечері або у вихідний день, не може цього зробити і змушений відкладати дію, що часто призводить до втрати ліда (потенційного покупця).

Приховані ціни та непрозорість ціноутворення є другою проблемою. На сторінках ресурсу прямо вказано: "Ціни на покупку комбікормів обумовлюються індивідуально з кожним покупцем. Дізнатися ціни... Ви можете зателефонувавши нам". Такий підхід створює високий психологічний бар'єр для нового клієнта та жорстко прив'язує його до графіка роботи відділу продажів. Наша розроблювана платформа, натомість, базується на концепції особистого кабінету: після авторизації клієнт автоматично бачить свою індивідуальну (персоналізовану) ціну, що формується на основі його партнерського статусу, умов договору чи обсягів закупівель.

Повна відсутність інтеграції зі складськими запасами є найчастішою проблемою більшості сайтів. Інтерфейс не надає жодної інформації про наявність товару. Клієнт не розуміє, чи може він розраховувати на відвантаження 5 тонн специфічного преміксу завтра вранці. Наша система передбачає впровадження модуля складського обліку, який транслює

актуальні залишки продукції на сайт у режимі реального часу, запобігаючи замовленням "у мінус".

Останньою проблемою є відсутність особистого кабінету (B2B порталу). Фермери та контрагенти не мають доступу до власного робочого простору. Вони не можуть переглянути історію своїх попередніх замовлень, щоб швидко "повторити покупку", позбавлені можливості самостійно завантажувати електронні видаткові накладні, сертифікати якості на партію товару чи акти звірок.

Наступним проаналізуємо сайт інтернет-магазину «Кормобаза».

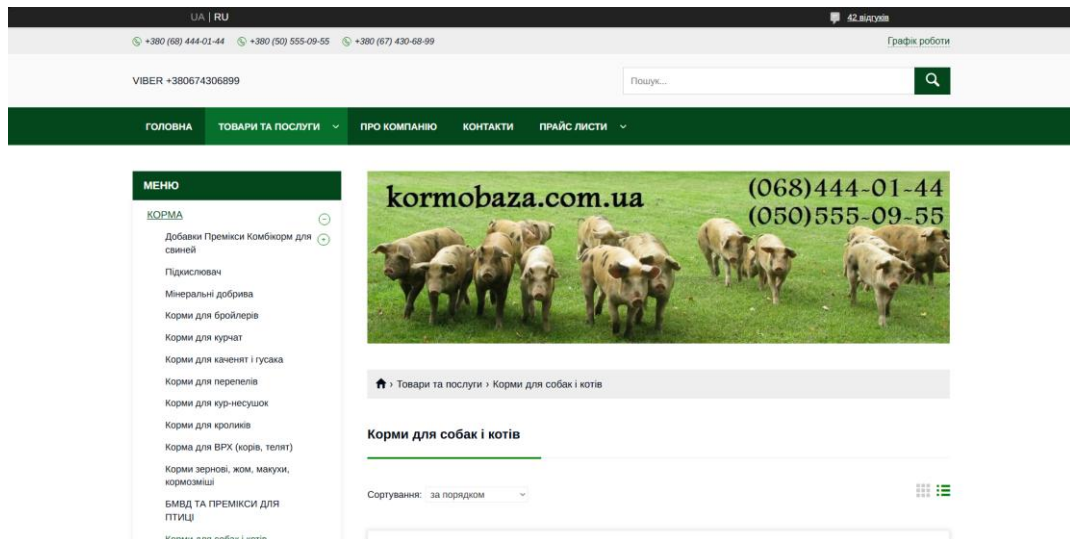


Рисунок 1.2 – Сторінка сайту «kormobaza.com.ua»

Цей ресурс є типовим інтернет-магазином, побудованим на базі стандартної CMS (Content Management System), і орієнтований переважно на роздрібного споживача (приватні домогосподарства). Він має базовий функціонал e-commerce: структурований каталог, відкриті ціни та кнопку "Купити". Проте архітектурно система зовсім не адаптована для обслуговування складних бізнес-процесів агропідприємства.

Відсутня B2B-логіка та матриці динамічного ціноутворення. Інтерфейс та логіка кошика оптимізовані виключно під купівлю 1-2 мішків корму (вказана роздрібна ціна за одиницю). Відсутня система автоматичного

перерахунку вартості залежно від обсягу замовленої партії (наприклад, при замовленні від 10 тонн). Для великих фермерських господарств оформлення оптових закупівель через такий інтерфейс є незручним і не відображає реальних комерційних умов співпраці.

Сайт використовує примітивні бінарні статуси товарів, характерні для базових інтернет-магазинів. Це не відображає реальної кількісної картини на складах виробництва. Виникає ризик "оверселлінгу": клієнт може додати до кошика 500 мішків, система успішно прийме замовлення, але фізично на складі їх може не бути. Наша розробка повинна включати механізм жорсткої валідації обсягів замовлення на основі поточних фізичних залишків та їх миттєвого резервування.

Також сайт має слабкий функціонал для менеджерів (відсутня ERP-елементів у бекенді). Подібні платформи зазвичай мають примітивну панель адміністратора, яка не здатна замінити систему внутрішнього обліку. Менеджерам доводиться працювати у двох системах: приймати замовлення на сайті, а потім вручну переносити дані (номенклатуру, суми, реквізити) у внутрішні облікові програми (наприклад, 1С або BAS). Це створює дублювання рутинної роботи та високий ризик помилок. Наша концепція передбачає об'єднання Frontend-вітрини для клієнта та Backend-обліку для працівників підприємства в єдину цілісну екосистему.

Сайт також перевантажений інформацією. Картки товарів на сайті здебільшого орієнтовані на пошукову оптимізацію (SEO) і містять масиви неструктурованого тексту. Для цільової аудиторії B2B сегмента (головних зоотехніків, ветеринарів, керівників ферм) критично важливо миттєво отримувати доступ до професійних характеристик. У розроблюваній системі дані про склад корму (сирий протеїн, обмінна енергія, рівень кальцію, фосфору, амінокислот) будуть винесені у чітко структуровані, стандартизовані таблиці для швидкого та зручного аналізу.

Проведений огляд конкурентів підтверджує, що існуючі на ринку рішення є або інформаційно обмеженими (корпоративні сайти-візитки), або

функціонально недостатніми для потреб середнього та великого агробізнесу (роздрібні B2C шаблони). Це обґрунтовує доцільність та актуальність розробки спеціалізованої комплексної вебплатформи, яка поєднає гнучкий інструментарій електронної комерції з надійною системою складського та управлінського обліку.

На основі проведеного аналізу існуючих конкурентів було сформовано порівняльну таблицю. Це дозволяє чітко окреслити вектор розробки та довести доцільність створення нового інформаційного продукту, який займе нішу між простим роздрібним магазином та корпоративним сайтом без функцій продажу.

Таблиця 1.1 – Порівняльна характеристика існуючих рішень та розроблюваної платформи

Функціональний критерій	«Кормобаза» (Шаблонний B2C магазин)	«Агрокапітал» (Корпоративний B2B сайт)	Пропонована вебплатформа
Онлайн оформлення замовлення	Присутнє (але тільки для роздрібу)	Відсутнє (тільки через дзвінок)	Повноцінний B2C та B2B кошик
Синхронізація залишків	Статична (ручне управління)	Відсутня	Real-time синхронізація (заборона замовлення відсутнього товару)
Особистий кабінет клієнта	Примітивний (лише історія покупок)	Відсутній	Розширений (персональні ціни, повтор замовлень, документообіг)
Динамічне ціноутворення	Ні (одна ціна для всіх)	Ні (ціни приховані, за запитом)	Автоматичний перерахунок залежно від об'єму та статусу клієнта
Внутрішній облік (Панель адміністратора)	Базовий функціонал CMS (лише контент)	Відсутній	Спеціалізована панель: управління партіями, складами, аналітичні звіти

Як видно з аналізу, розроблюваний проект має закрити існуючу прогалину на ринку. Пропонована інформаційна система виступає як комплексне B2B/B2C рішення, що об'єднує зручність онлайн-замовлень для

клієнтів (Frontend) з потужним модулем товарного обліку для адміністраторів підприємства (Backend).

### 1.3 Постановка задачі

Метою створення інформаційної вебплатформи для агропромислового підприємства є розробка сучасної цифрової B2B/B2C системи, яка забезпечить ефективний продаж комбикормів в інтернет-середовищі, покращить комунікацію з клієнтами та дозволить зручно управляти товарними залишками, замовленнями і зворотним зв'язком через адміністративну панель.

Вебзастосунок має стати не лише інструментом онлайн-продажів та зручної демонстрації асортименту продукції, але й надійним каналом залучення нових контрагентів, підвищення рівня клієнтського сервісу та автоматизації внутрішніх складських і логістичних процесів підприємства.

Цілі проєкту:

1. Створити інтерактивний вебзастосунок (за архітектурою Single Page Application), який забезпечить підприємству ефективну цифрову присутність та цілодобовий доступ клієнтів до актуального каталогу продукції.

2. Розробити комплексну адміністративну панель, що дозволить персоналу легко додавати нові товари, контролювати наявність кормів на складі та управляти статусами замовлень в єдиному інтерфейсі.

3. Забезпечити зручну взаємодію з користувачами шляхом впровадження системи особистих кабінетів, де покупці зможуть відстежувати історію своїх покупок, перевіряти статуси доставки та надсилати звернення до адміністрації.

4. Реалізувати адаптивний дизайн та гнучку систему багатофакторної фільтрації (наприклад, за типом тварин: кури, свині, ВРХ, гуси), що гарантує швидкий та безпомилковий пошук необхідних кормів з будь-якого пристрою [4].

5. Впровадити модуль аналітики (дашборди) для керівництва підприємства, що дозволить у режимі реального часу відстежувати фінансові показники, кількість замовлень та популярність окремих товарних позицій.

6. Забезпечити надійні механізми автономного локального збереження інформації з можливістю швидкого експорту даних (реєстри користувачів, списки замовлень, складські запаси) у формат CSV для подальшого аналізу та синхронізації.

7. Створити масштабовану архітектуру на базі сучасних вебтехнологій (HTML, CSS, JavaScript) з чітким розмежуванням прав доступу на основі ролей (адміністратор, клієнт) для гарантування безпеки даних.

#### 1.4 Висновки до розділу 1

Проведений аналіз предметної області показав, що в умовах сучасного ринку головною перешкодою для ефективного розвитку підприємств з виробництва комбікормів є використання застарілих, фрагментованих методів обліку готової продукції та ручної обробки замовлень. Відсутність єдиної цифрової платформи призводить до розсинхронізації даних, логістичних помилок через «людський фактор» та неможливості цілодобового (24/7) обслуговування клієнтів.

Аналіз існуючих на ринку аналогів, таких як корпоративний сайт «Агрокапітал» та шаблонний роздрібний інтернет-магазин «Кормобаза», виявив їхні фундаментальні архітектурні та функціональні недоліки. Досліджені ресурси не забезпечують інтеграції зі складськими запасами в режимі реального часу, позбавлені алгоритмів динамічного ціноутворення для B2B-сегмента та не надають контрагентам доступу до повноцінного особистого кабінету. Це беззаперечно підтверджує науково-практичну актуальність та доцільність створення нового спеціалізованого програмного продукту, що закrije існуючу прогалину на ринку.

На основі виявлених операційних проблем підприємства та недоліків конкурентів було сформовано постановку задачі та визначено ключові цілі проєкту. Розроблювана система має функціонувати як сучасний інтерактивний вебзастосунок (Single Page Application), що органічно поєднує інструменти електронної комерції для клієнтів (B2B/B2C) із потужною адміністративною панеллю. Платформа повинна забезпечити гнучку фільтрацію товарів, управління замовленнями, аналітику продажів через дашборди, а також механізми автономного локального збереження даних з експортом у CSV.

Таким чином, розробка власної інтегрованої вебплатформи з використанням сучасних вебтехнологій (HTML, CSS, JavaScript) є стратегічно важливим кроком. Впровадження такого рішення дозволить підприємству ліквідувати проблему розрізненого обліку, підвищити пропускну здатність відділу збуту, покращити клієнтський сервіс та розширити ринки збуту продукції.

## РОЗДІЛ 2 ІНФОРМАЦІЙНЕ ТА МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

### 2.1 Аналіз предметної області

Предметна область агропромислового комплексу (АПК), зокрема в сегменті виробництва та реалізації комбінованих кормів, є надзвичайно складною, багатокomпонентною та динамічною сферою [5]ку. Сьогодні підприємства цього профілю функціонують як високотехнологічні комплекси, діяльність яких має специфічні закономірності розвитку та значний перелік взаємопов'язаних бізнес-процесів. Для створення інформаційної вебплатформи, здатної ефективно автоматизувати діяльність такого підприємства, необхідно здійснити детальний аналіз предметної області. Це формує фундамент для подальшого проєктування системи, визначення її функціональних вимог, структури баз даних та загальної архітектури. Аналіз дозволяє виявити ключові елементи галузі, типові проблеми, такі як розсинхронізація даних, та потреби, які розв'язуються за допомогою цифрових B2B/B2C рішень.

Процес виробництва, обліку та реалізації комбікормів передбачає взаємодію численних учасників: менеджерів з продажу, працівників складів, керівництва підприємства, а також широкого спектра клієнтів. Клієнтська база є вкрай неоднорідною та поділяється на великий B2B сегмент (птахофабрики, агрохолдинги) та дрібний B2B/B2C сегмент (локальні фермери, приватні домогосподарства). Для кожного з цих учасників виникає необхідність доступу до різних типів інформації: технічної (склад кормів), фінансової (персоналізовані ціни) та логістичної (наявність на складі). Тому будь-яка цифрова система повинна не лише враховувати багатоаспектність цієї сфери, але й бути здатною забезпечувати надійний, швидкий та структурований обіг даних між усіма контрагентами.

Особливістю аграрної сфери є те, що більшість її процесів мають яскраво виражений сезонний та циклічний характер. Асортимент продукції, що включає повнораціонні комбікорми, премікси та білково-мінерально-вітамінні добавки, постійно оновлюється залежно від наявної сировини та вікових потреб сільськогосподарських тварин у різні пори року. Відповідно, предметна область характеризується високою номенклатурною варіативністю, що створює додаткові вимоги до автоматизації: система повинна бути достатньо гнучкою, адаптивною та здатною до миттєвого оновлення даних у каталогах.

У межах предметної області варто окремо виділити етап взаємодії з замовником, оскільки саме він є ключовою точкою генерації прибутку. Традиційні підходи з ручним прийомом заявок через телефон або месенджери створюють жорсткі «вузькі місця» та підвищують ризик операційних помилок, коли менеджер може переплутати рецептуру корму (наприклад, «Старт» і «Фінішер») або адресу доставки. Цифрова платформа повинна кардинально змінити цей процес, перевівши клієнтів на формат самообслуговування 24/7 з можливістю самостійного формування замовлень у кошику. Це вимагає від системи надійної фіксації вимог користувача та забезпечення прозорої комунікації на всіх етапах обробки заявки.

Важливим аспектом галузі є складна, багатофакторна матриця ціноутворення. Кінцева вартість продукції для контрагента формується динамічно і залежить від обсягу замовленої партії (роздріб чи вагонна норма), історії співпраці, персональних накопичувальних знижок та умов договору. Ця фінансова інформація постійно змінюється, тому вебплатформа повинна мати механізм автоматичної ідентифікації клієнта після авторизації та миттєвого відображення його персоналізованих цін. Це вимагає високого рівня точності, що неможливо забезпечити без належної інформаційної підтримки бази даних.

Не менш складною є складська та логістична складова. Агропідприємство оперує величезними обсягами сировини та готової продукції. Проблемою предметної області є фрагментація інформаційного

простору: дані про залишки часто зберігаються у закритих системах (наприклад, 1С), тоді як продажі ведуться розрізнено. Це призводить до ризику «оверселлінгу» (продажу відсутнього товару) та зриву строків постачання. Отже, система автоматизації повинна забезпечувати механізми жорсткої валідації обсягів замовлень на основі поточних фізичних залишків на складі та їх миттєвого резервування у режимі реального часу.

Окрему роль у предметній області відіграє документообіг та управління реєстрами. Діяльність виробника комбікормів супроводжується значною кількістю даних: реєстрами клієнтів, історією замовлень, інформацією про наявність продукції на складах. Суттєвою потребою є забезпечення механізмів зберігання, пошуку, сортування та швидкого експорту цих даних у формати на кшталт CSV для подальшого аналізу чи синхронізації з бухгалтерськими програмами. Це мінімізує операційні затримки та підвищує загальну керованість підприємства.

Крім основних процесів продажу, у предметній області агропромисловості критично важливими є питання зворотного зв'язку та контролю якості. Будь-яка вироблена партія корму повинна відповідати стандартам, а відхилення можуть призвести до захворюваності тварин у замовника. Тому вебзастосунок повинен забезпечити централізовану систему комунікації між виробником та кінцевим споживачем. Впровадження системи тикетів (звернень) в особистому кабінеті дозволяє клієнтам фіксувати проблеми з логістикою, повідомляти про якість продукції та отримувати оперативні відповіді від адміністрації, формуючи єдиний і прозорий простір взаємодії.

Важливим аспектом також є потреба у формуванні аналітичної звітності для прийняття Data-Driven рішень. У ручному режимі керівництво підприємства часто позбавлене можливості оперативно відстежувати динаміку продажів за конкретними товарними категоріями чи групами тварин. Автоматизована система повинна мати модуль аналітики (дашборди), що

дозволяє у режимі реального часу відстежувати фінансові показники, популярність позицій та стан критичних залишків на складах.

Таким чином, предметна область агропромислового підприємства з виробництва комбікормів є багатогранною, складною та взаємопов'язаною екосистемою. Ретельний аналіз цього середовища доводить, що просте розширення штату працівників не здатне вирішити існуючі комунікаційні та логістичні проблеми. Це дає змогу сформулювати повне розуміння потреб у цифровій підтримці, що надалі стає основою для проектування архітектури системи, яка органічно об'єднає функції електронної комерції (Frontend) з бекенд-інструментами управління складом та ресурсами (ERP/WMS).

## 2.2 Моделювання бізнес-процесів агропромислового підприємства

Моделювання бізнес-процесів є фундаментальним етапом у розробці інформаційної вебплатформи, особливо у контексті агропромислового комплексу [6], де ефективність реалізації продукції безпосередньо впливає на рентабельність підприємства. Для підприємства, що спеціалізується на виробництві комбікормів для різних видів сільськогосподарських тварин та птиці, ключовим завданням є оптимізація комунікацій із гуртовими та роздрібними покупцями, забезпечення прозорості обліку продукції та створення зручного інструментарію для управління онлайн-замовленнями. У цьому контексті вебзастосунок виконує роль не просто електронного каталогу, а повноцінної системи управління взаємовідносинами з клієнтами (CRM) та облікової платформи, що дозволяє систематизувати бізнес-процеси та зробити їх більш контрольованими.

Одним із основних напрямків моделювання є аналіз взаємодії користувача (фермера, представника агрогосподарства або приватного покупця) із вебплатформою. Цей шлях починається від відвідування головної сторінки, переходу до каталогу продукції, використання специфічних фільтрів та завершується оформленням замовлення. Головна сторінка виконує функцію

первинної навігації, де користувач отримує базову інформацію про виробника, асортимент та умови співпраці. У науковому підході до моделювання цього процесу передбачено деталізацію всіх можливих дій користувача. Особлива увага приділяється механізму фільтрації: оскільки підприємство виготовляє корми для різних видів тварин, клієнт повинен мати можливість швидко відсортувати продукцію за цільовими групами (наприклад, корми для курей, гусей, свиней, великої рогатої худоби).

Ключовим бізнес-процесом на стороні клієнта є формування та обробка онлайн-замовлень. Кожне замовлення формується через віртуальний кошик, куди користувач додає необхідні види комбікормів із зазначенням об'єму (ваги) та типу фасування. Моделювання цього процесу передбачає виділення окремих логічних етапів: авторизація клієнта, вибір продукції, перевірка актуальної ціни, підтвердження контактних даних та адреси доставки, і, зрештою, фіксація замовлення у базі даних. Важливо враховувати потенційні альтернативні сценарії, такі як відсутність певного товару на складі або введення некоректних контактних даних, що вимагає імплементації надійних алгоритмів валідації на стороні клієнтського інтерфейсу.

Для візуалізації структури даних та взаємозв'язків між основними сутностями системи на етапі клієнтської взаємодії розроблено UML-діаграму класів (рис. 2.1).

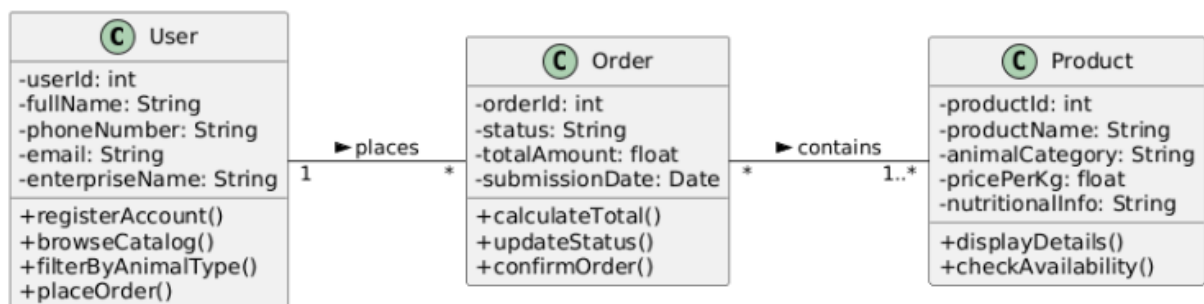


Рисунок 2.1 – UML-діаграма класів взаємодії користувача із каталогом та замовленнями

Окремої та глибокої уваги під час моделювання заслуговує адміністративна панель вебплатформи, яка виступає центральним вузлом управління всіма даними підприємства. Моделювання роботи адміністратора кардинально відрізняється від клієнтського сценарію, оскільки вимагає реалізації складнішої бізнес-логіки, орієнтованої на аналітику, облік та управління персональними даними.

Аналіз вимог до адміністративної частини виявив необхідність впровадження багатофункціонального дашборду. Замість традиційного і часто неінформативного відображення загальних залишків на складах, логіка дашборду сфокусована на аналітиці продажів – системі необхідно пріоритетно виводити статистику щодо найпопулярніших видів кормів, які замовляються клієнтами найчастіше. Це дозволяє керівництву підприємства оперативно реагувати на ринковий попит та своєчасно коригувати виробничі плани.

Крім того, моделювання бізнес-процесу адміністрування включає управління користувачами та їхнім зворотним зв'язком. Адміністратор системи повинен мати інструментарій для глибокого редагування профілів користувачів (наприклад, оновлення реквізитів агрокомпаній, зміна статусів гуртових клієнтів), а також управління історією відгуків (модерація, відповіді на запити, аналіз рекламацій щодо якості продукції).

Важливою архітектурною та функціональною вимогою до системи є можливість експорту даних. Оскільки значна частина обліку в агропромисловому секторі традиційно ведеться у локальних електронних таблицях, бізнес-процес обробки замовлень передбачає генерацію та вивантаження даних у форматі CSV. Цей процес охоплює формування файлів із чітко визначеними стовпцями (дата, номенклатура, об'єм, контрагент), що дозволяє легко інтегрувати дані з вебплатформи у внутрішні бухгалтерські системи підприємства або локальні Excel-документи.

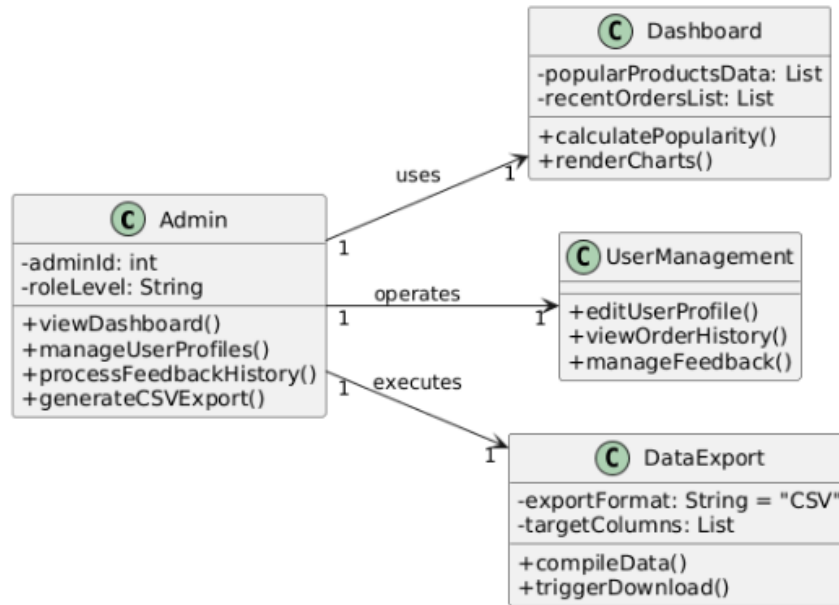


Рисунок 2.2 – UML-діаграма класів адміністративної панелі та управління даними

Моделювання загальної взаємодії у системі найдоцільніше подати у вигляді діаграми варіантів використання (Use Case). Ця модель візуалізує функціональні можливості платформи з точки зору різних акторів: «Клієнта» (користувача) та «Адміністратора» (менеджера підприємства). Для клієнта критично важливими прецедентами є навігація по каталогу, фільтрація комбікормів за специфікацією (птиця, худоба), оформлення замовлення та залишення відгуку про співпрацю.

Зі свого боку, адміністратор має розширений набір прецедентів, що відповідають його повноваженням. Серед них: аналіз популярної продукції на дашборді, що дозволяє оптимізувати виробництво; редагування профілів клієнтів для підтримання актуальної бази контрагентів; управління історією відгуків для контролю репутації; та виконання експорту даних замовлень у CSV-формат для подальшої обробки у зовнішніх програмах локального обліку.

Науковий підхід до моделювання цих прецедентів гарантує, що розроблена архітектура покриватиме всі реальні потреби агропромислового

підприємства, уникаючи надлишкового функціоналу та зосереджуючись на тих інструментах, які дійсно приносять цінність бізнесу.



Рисунок 2.3 – UML-діаграма варіантів використання інформаційної вебплатформи

Процес моделювання також дозволяє передбачити альтернативні та виняткові сценарії, такі як відмова у завантаженні бази даних, спроба експорту занадто великого масиву даних адміністратором, або введення клієнтом некоректних параметрів при фільтрації продукції. Детальне опрацювання цих аспектів на етапі бізнес-моделювання дозволяє закласти надійні алгоритми обробки винятків у майбутній програмний код, забезпечуючи високу стабільність та відмовостійкість системи в умовах реального виробничого навантаження.

### 2.3 Висновки до розділу 2

У розділі 2 було проведено глибоке дослідження інформаційного середовища та бізнес-процесів агропромислового підприємства, що спеціалізується на виробництві комбікормів. На початковому етапі

проаналізовано предметну область, що дозволило ідентифікувати ключових суб'єктів взаємодії (гуртових і роздрібних клієнтів, адміністративний персонал) та визначити специфіку їхніх інформаційних потреб. Дослідження підтвердило, що висока номенклатурна варіативність продукції, сезонність попиту та необхідність динамічного ціноутворення вимагають впровадження гнучких інструментів автоматизації. Виявлено, що розсинхронізація даних між закритими системами обліку та каналами продажів є критичною проблемою, яка призводить до логістичних помилок та зниження ефективності обслуговування.

Наступним кроком стало моделювання бізнес-процесів підприємства з фокусом на оптимізацію взаємодії користувачів із системою. Побудовано UML-діаграму класів для клієнтської частини, яка формалізувала структуру даних, необхідних для формування онлайн-замовлень, включаючи інформацію про користувача, склад кошика та параметри продукції. Це забезпечило логічну основу для реалізації зручного каталогу з функцією багатофакторної фільтрації.

Окрему увагу приділено проєктуванню адміністративної панелі. UML-моделювання її архітектури виявило потребу у створенні аналітичних дашбордів для відстеження популярних товарних позицій, інструментів управління профілями клієнтів та модулів для роботи зі зворотним зв'язком. Важливим етапом стало визначення механізмів експорту даних, що дозволить інтегрувати вебплатформу з існуючими бухгалтерськими програмами через CSV-формат.

Завершальним етапом аналітичної роботи стала побудова UML-діаграми варіантів використання, яка наочно продемонструвала розподіл функцій між клієнтами та адміністраторами. Ця модель підтвердила повноту та несуперечливість закладеного функціоналу, гарантуючи, що спроектована система задовольнятиме всі ключові потреби агробізнесу.

Проведені в цьому розділі дослідження та моделювання створили надійне інформаційне та логічне підґрунтя для наступного етапу – програмної

реалізації вебплатформи. Визначені сутності, їхні атрибути та зв'язки стали основою для проєктування структури збереження даних та розробки клієнтського і серверного функціоналу. Впровадження автоматизованої системи дозволить вирішити проблему розрізненого обліку, підвищити швидкість обробки замовлень та покращити якість клієнтського сервісу.

## РОЗДІЛ 3 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

### 3.1 Проектування та реалізація структури збереження даних

Фундаментальним етапом створення будь-якої сучасної інформаційної вебплатформи є розробка архітектури збереження, обробки та управління даними. Для агропромислового підприємства, що спеціалізується на виробництві та реалізації комбікормів, забезпечення цілісності, швидкодоступності та безпеки інформації є критичним фактором успішного функціонування. Інформаційна система повинна безперервно обробляти значні масиви даних, що включають номенклатуру готової продукції, реєстри контрагентів, історію замовлень та потоки зворотного зв'язку.

Враховуючи специфіку даного проекту, який орієнтований на локальне виконання у середовищі браузера без залучення складних та ресурсомістких серверних баз даних, архітектурне рішення базується на використанні технології Web Storage API, зокрема синхронного сховища localStorage. Цей підхід дозволяє зберігати інформацію безпосередньо у браузері користувача у вигляді пар "ключ-значення"[7], де всі складні структури даних серіалізуються у текстовий формат JSON [8]. На відміну від класичних реляційних баз даних, такий документо-орієнтований підхід забезпечує максимальну швидкість розгортання платформи, ідеальну сумісність із JavaScript-логікою застосунку та повну автономність роботи клієнтської частини.

Проектування структур даних здійснювалося з глибоким урахуванням вимог до адміністративної панелі та бізнес-логіки підприємства. Сучасний агробізнес вимагає гнучких інструментів аналітики. Аналітичний дашборд керівника має фокусуватися не просто на констатації загальних залишків сировини на складах, а насамперед на виявленні та візуалізації даних щодо найпопулярніших видів продукції. Пріоритезація аналітики продажів популярних кормів дозволяє підприємству оптимізувати виробничі

потужності, вчасно закуповувати сировину для найбільш рентабельних позицій (наприклад, для кормів для курей, свиней чи великої рогатої худоби) та мінімізувати економічні втрати від надвиробництва неліквідних товарів. Для забезпечення цієї логіки, в об'єктах продукції передбачено спеціальний прапорець `popular`, що дозволяє системі миттєво ідентифікувати та виводити на головну сторінку ключові позиції сезону.

Крім того, адміністративний модуль вимагає надійних інструментів для управління профілями контрагентів та ведення історії зворотного зв'язку. Архітектура даних передбачає наявність колекції `db_feedback`, яка зберігає всі запити користувачів, їхні причини та статуси обробки адміністратором. Це дозволяє створити повноцінну систему підтримки, де кожне звернення надійно прив'язується до конкретного користувача через його унікальний ідентифікатор, забезпечуючи прозорість комунікації та можливість ретроспективного аналізу якості обслуговування.

Невіддільною вимогою до системи збереження даних є інтероперабельність – здатність системи обмінюватися даними із зовнішніми бухгалтерськими та управлінськими програмами, які традиційно використовуються в агросекторі. Зважаючи на це, структура JSON-об'єктів спроектована таким чином, щоб забезпечити безшовний та повністю автоматизований експорт даних у формат CSV (Comma-Separated Values). Реалізований механізм дозволяє адміністратору в один клік вивантажувати реєстри продукції, клієнтів, замовлень та відгуків у вигляді структурованих таблиць, повністю готових для імпорту в сторонні аналітичні системи. Кожне поле JSON-об'єкта жорстко відповідає окремій колонці у згенерованому CSV-файлі, що унеможлиблює втрату або спотворення даних під час експорту.

Першим і фундаментальним сховищем у розробленій системі є колекція `db_users`, яка виконує роль централізованого реєстру всіх контрагентів та адміністративного персоналу платформи. Від коректної організації цієї колекції залежить функціонування модуля авторизації, безпека персональних даних, можливість редагування профілів, розмежування прав доступу та

загальна стабільність платформи. Колекція ініціалізується у вигляді масиву об'єктів, де кожен об'єкт репрезентує окремого користувача.

Структура сутності користувача розроблена з урахуванням необхідності зберігання вичерпної контактної та ідентифікаційної інформації. Розглянемо детально кожне поле колекції `db_users`:

Першим ключовим полем є `username`. Воно має рядковий тип даних і виконує функцію унікального ідентифікатора (логіна) в межах усієї системи. На відміну від числових ідентифікаторів, використання логіна є більш зрозумілим для користувачів. Це поле є первинним ключем для зв'язку користувача з його замовленнями та історією відгуків. Система жорстко контролює унікальність цього поля під час реєстрації нового кабінету, запобігаючи створенню дублікатів. Прикладом значення може бути "admin" або "ivanov".

Наступним є поле `name`, яке також відноситься до рядкового типу. Воно призначене для збереження повного найменування контрагента. Оскільки цільовою аудиторією вебплатформи є як приватні особи, так і фермерські господарства, це поле може містити як стандартне прізвище, ім'я та по батькові, так і офіційну назву підприємства (наприклад, "ФГ 'Агро-Лідер' (Іванов П.М.)"). Ця інформація є критично важливою для персоналізації комунікації в особистому кабінеті та забезпечення можливості редагування профілю адміністратором.

Поле `email` відповідає за збереження електронної адреси користувача. Електронна пошта є не лише запасним каналом зв'язку з клієнтом, але й стандартним реквізитом для ведення ділової переписки в гуртовому сегменті. Прикладом такого запису є "admin@agroprom.ua" або "ivanov@gmail.com". Наявність цього поля в базі даних забезпечує гнучкість контактної інформації.

Для забезпечення оперативного зв'язку логістичної служби підприємства з клієнтами передбачено поле `phone`. Контактний номер телефону є найважливішим елементом комунікації під час узгодження деталей відвантаження та доставки комбікормів. Зважаючи на можливі варіації

форматів запису номерів, поле зберігає дані у текстовому вигляді (наприклад, "+38 (067) 555-44-33"), що дозволяє включати коди країн та форматуючі символи.

Питання безпеки доступу вирішується за допомогою поля password. У даній реалізації воно зберігає пароль доступу до облікового запису. Рядок використовується для перевірки автентичності під час входу в систему (наприклад, "user123" або "admin"). Архітектура системи дозволяє користувачам самостійно оновлювати це поле через інтерфейс особистого кабінету.

Останнім, але найбільш значущим з точки зору архітектури, є поле role. Це поле виконує функцію класифікатора рівня доступу суб'єкта до функціонала платформи. Система підтримує рольову модель доступу, розрізняючи два основні стани: "client" для звичайних покупців та "admin" для персоналу підприємства. Наявність значення "admin" у цьому полі розблоковує приховані елементи навігації та надає доступ до адміністративного пульта керування, аналітичних дашбордів, реєстрів користувачів та інструментів експорту CSV. Значення "client", своєю чергою, обмежує права користувача виключно його особистим кабінетом та власними замовленнями.

Візуальне представлення структури колекції користувачів, із зазначенням типів даних, описом бізнес-логіки та прикладами реальних записів, наведено у таблиці 3.1.

Таблиця 3.1 – Структура та опис колекції db\_users

Назва атрибута	Тип даних	Опис та бізнес-логіка	Приклад значення
username	String	Унікальний ідентифікатор (логін) в системі.	"ivanov"
name	String	Повне найменування компанії або ПІБ фізичної особи.	"ФГ 'Агро-Лідер' (Іванов П.М.)"

email	String	Контактна електронна адреса контрагента.	"ivanov@gmail.com"
phone	String	Контактний телефон для логістичного зв'язку.	"+38 (067) 555-44-33"
password	String	Пароль для проходження процедури авторизації.	"user123"
role	String	Рівень прав доступу, що регулює видимість адміністративної панелі.	"client"

Другим базовим компонентом інформаційної структури вебплатформи є колекція `db_orders`, яка функціонує як централізований журнал реєстрації та обліку онлайн-замовлень агропромислового підприємства. Цей масив об'єктів є ключовою ланкою всього операційного циклу системи, оскільки він забезпечує збереження комерційної історії взаємодії з клієнтами, фіксує обсяги реалізованої продукції та дозволяє адміністрації здійснювати моніторинг поточного стану виконання кожної транзакції.

Кожен об'єкт у складі масиву `db_orders` містить фіксований набір атрибутів, що вичерпно описують параметри комерційної угоди. На основі аналізу архітектури вебплатформи визначено такі структурні поля:

Першим атрибутом виступає `id`, що виконує функцію унікального коду замовлення в системі. Цей ідентифікатор автоматично присвоюється кожній новій заявці в момент її підтвердження користувачем. Наявність унікального числового або рядкового коду є фундаментальною вимогою для ведення внутрішнього обліку, швидкого пошуку транзакцій адміністратором, а також для автоматизованого вивантаження звітів, де кожна операція повинна мати чітку мітку.

Поле `username` відноситься до рядкового типу і містить логін контрагента, який оформив дане замовлення. Це поле забезпечує логічний зв'язку між колекцією замовлень та реєстром користувачів `db_users`. Завдяки

такій архітектурній прив'язці система може динамічно відображати історію покупок в особистому кабінеті конкретного клієнта, а адміністратор має можливість аналізувати купівельну спроможність та активність окремих фермерських господарств.

Хронологічна фіксація операцій реалізується через атрибут `date`. Це рядкове поле зберігає точну дату та час створення замовлення. Дані цього поля використовуються для сортування журналу операцій, ретроспективного аналізу динаміки продажів комбікормів за певні проміжки часу (тижні, місяці, сезони) та оптимізації календарного планування логістичних потоків підприємства.

Для збереження детальної інформації про склад придбаної продукції призначене поле `items`. Воно містить структуровані дані у вигляді переліку ідентифікаторів товарів та їхньої кількісної мірки. Цей атрибут дозволяє складським працівникам точно ідентифікувати, які саме види комбікормів та в якому обсязі необхідно підготувати до відвантаження за конкретною заявкою.

Фінансова складова транзакції описується числовим атрибутом `total`, що фіксує сумарну вартість усього переліку замовлених товарів у гривнях. Значення цього поля розраховується автоматично на основі поточних цін комбікормів та їхньої кількості в кошику. Дані з цього атрибута є фундаментальними для формування фінансової звітності та аналітики загального прибутку підприємства, а також експортуються до бухгалтерських модулів через CSV-файли.

Останнім керуючим атрибутом є `status`, який має рядковий тип даних. Це поле відображає поточний етап життєвого циклу замовлення (наприклад, "В обробці", "Підтверджено", "Відвантажено", "Виконано"). Зміна значення цього атрибута здійснюється адміністратором через панель управління, що дозволяє клієнту в реальному часі спостерігати за процесом підготовки його партії комбікорму в особистому кабінеті.

Візуальне представлення архітектури колекції замовлень, із зазначенням системних назв атрибутів, відповідних типів даних та описів бізнес-логіки, наведено у таблиці 3.2.

Таблиця 3.2 – Структура та опис колекції db\_orders

Назва атрибута	Тип даних	Опис та бізнес-логіка	Приклад значення
id	Number / String	Унікальний системний ідентифікатор замовлення.	20045
username	String	Логін клієнта, що забезпечує зв'язок із профілем користувача.	"ivanov"
date	String	Текстова мітка дати та часу оформлення транзакції.	"2026-06-15 "
items	String	Перелік замовлених позицій комбікормів та їхньої кількості.	"ПК-1 (200 кг)"
total	Number	Загальна вартість замовлення у національній валюті (грн).	6750
status	String	Поточний операційний стан обробки та відвантаження замовлення.	"В обробці"

Наступною, не менш важливою складовою архітектури даних вебплатформи, є колекція db\_products. Цей масив об'єктів виконує функцію електронного каталогу товарів та одночасно слугує віртуальним відображенням стану складських запасів агропромислового підприємства. Саме до цієї колекції звертаються алгоритми клієнтської частини для формування асортименту на вітрині, розрахунку вартості в кошику та перевірки наявності продукції під час оформлення заявок покупцями.

Кожен елемент масиву db\_products є структурованим JSON-об'єктом, що містить вичерпні фізичні, економічні та логічні характеристики конкретного

виду комбікорму чи білково-мінеральної вітамінної добавки (БМВД). Розглянемо детально призначення кожного атрибута:

Атрибут `id` є числовим ідентифікатором товарної позиції. Він автоматично генерується системою під час додавання нового продукту адміністратором через панель керування. Цей код використовується як первинний маркер для точної ідентифікації товару алгоритмами кошика та при формуванні фінального списку замовлених позицій.

Поле `title` має рядковий тип даних і містить повну комерційну та номенклатурну назву корму. Ця текстова інформація виводиться у каталозі, таблицях кошика та експортованих звітах, дозволяючи контрагентам та працівникам складу чітко розуміти специфікацію товару.

Економічна складова товару фіксується у полі `price`, яке зберігає числове значення базової вартості одного кілограма продукції у національній валюті. Використання виключно числового формату для цього поля є критичною умовою для безпомилкової роботи математичних функцій платформи, що динамічно обчислюють загальну суму покупок.

Для забезпечення жорсткого контролю за складськими залишками впроваджено числове поле `stock`. Воно відображає поточну наявну вагу конкретного корму на фізичному складі підприємства у кілограмах. Бізнес-логіка платформи налаштована таким чином, що система блокує можливість додати до кошика обсяг продукції, який перевищує значення цього атрибута, а після підтвердження замовлення залишок автоматично декрементується.

Поле `tags` реалізоване як масив рядків і є основою системи пошуку та категоризації. Тут зберігаються ключові слова-маркери (наприклад, "кури", "свині", "добавки"), на які спирається модуль фільтрації в каталозі. Таке рішення дозволяє швидко сортувати асортимент за призначенням для конкретного виду сільськогосподарських тварин.

Візуальна презентація товару в інтерфейсі оптимізована за допомогою рядкового атрибута `img`. З метою економії ресурсів локального сховища, замість повноцінних зображень тут зберігається системний ідентифікатор

категорії пакування (наприклад, "poultry", "cow", "additive"). На основі цього маркера скрипти автоматично генерують відповідне кольорове тло та іконки для карток товарів.

Маркетингова гнучкість платформи забезпечується логічним (булевим) атрибутом popular. Якщо це поле набуває значення true, товар ідентифікується як хіт продажу та алгоритмічно дублюється у спеціальному промо-блоці на головній сторінці, що дозволяє акцентувати увагу покупців на найбільш актуальних сезонних пропозиціях.

Детальний опис структури колекції складської номенклатури наведено у таблиці 3.3.

Таблиця 3.3 – Структура та опис колекції db\_products

Назва атрибута	Тип даних	Опис та бізнес-логіка	Приклад значення
id	Number	Унікальний числовий ідентифікатор товару в базі.	201
title	String	Повна назва комбікорму або кормової добавки.	"Комбікорм ПК-1 (Повнораціонний для курей-несучок)"
price	Number	Вартість одного кілограма продукції (грн).	14.50
stock	Number	Поточний доступний залишок товару на складі підприємства (кг).	4800
tags	Array of Strings	Масив тегів для роботи системи фільтрації за призначенням.	["кури"]
img	String	Кодовий маркер для генерації візуального стилю пакування в інтерфейсі.	"poultry"
popular	Boolean	Логічний прапорець для виведення товару на головну сторінку як хіта продажів.	true

Останнім структурним масивом даних, що забезпечує комунікаційну взаємодію та повноцінне функціонування клієнтського сервісу вебплатформи, є колекція `db_feedback`. Вона виконує функцію електронного журналу реєстрації технічної підтримки, скарг, відгуків та комерційних пропозицій від контрагентів. Впровадження цієї колекції в архітектуру системи дозволяє адміністрації агропромислового підприємства централізовано акумулювати зворотний зв'язок, оперативно реагувати на логістичні запити та вести історію діалогів із кожним окремим клієнтом.

Кожен об'єкт у цій колекції репрезентує окреме звернення (тікет) та містить набір полів, що вичерпно описують суть проблеми та етап її вирішення:

Системне поле `id` містить унікальний числовий ідентифікатор тікета. Цей номер автоматично генерується під час відправлення форми зворотного зв'язку з особистого кабінету користувача. Наявність ідентифікатора є необхідною для точної адресації відповідей адміністратором та ведення архівного обліку звернень.

Атрибут `username` (рядковий тип) забезпечує реляційний зв'язок звернення із конкретним профілем контрагента з колекції `db_users`. Завдяки цьому полю система здатна ідентифікувати автора повідомлення та коректно виводити історію листування в його персональному кабінеті.

Для класифікації вхідних запитів використовується поле `reason`. Воно зберігає текстову категорію звернення (наприклад, "Проблема з доставкою", "Якість продукції", "Помилка в розрахунках"). Цей атрибут дозволяє адміністративному персоналу швидко оцінювати тематику повідомлень та пріоритетувати їхню обробку.

Безпосередній зміст звернення фіксується у текстовому полі `message`. Тут зберігається розгорнутий опис проблеми або пропозиції, власноруч сформульований клієнтом під час створення тікета.

Для контролю за дотриманням термінів надання підтримки передбачено атрибут `date`, який зберігає точну дату ініціації звернення у рядковому

форматі. Це дозволяє формувати хронологічно правильну стрічку активності в адміністративній панелі.

Зворотна комунікація з боку підприємства реалізується через поле reply. До моменту опрацювання запиту це поле залишається порожнім. Після розгляду звернення сюди записується офіційна текстова відповідь адміністратора, яка миттєво стає доступною для перегляду в кабінеті клієнта.

Останнім є керуюче поле status. Цей рядковий індикатор відображає поточний стан життєвого циклу тикета. Базовими станами є "open" (нове звернення, що очікує розгляду) та "answered" (запит успішно опрацьовано та надано відповідь).

Детальний опис структури колекції зворотного зв'язку наведено у таблиці 3.4.

Таблиця 3.4 – Структура та опис колекції db\_feedback

Назва атрибута	Тип даних	Опис та бізнес-логіка	Приклад значення
id	Number	Унікальний системний ідентифікатор звернення (тикета).	901
username	String	Логін клієнта-ініціатора для прив'язки до профілю.	"ivanov"
reason	String	Категорія запиту для сортування в адміністративній панелі.	"Проблема з доставкою"
message	String	Основний текст скарги, питання або комерційної пропозиції.	"Чи можна узгодити доставку зерновозом на вівторок?"
date	String	Хронологічна мітка створення звернення.	"2026-06-11"
reply	String	Текстова відповідь адміністратора підприємства.	"Так, замовлення призначено на спецтранспорт."
status	String	Індикатор стану обробки звернення	«answered»

### 3.2 Програмна реалізація

Практичний етап розробки інформаційної вебплатформи «АгроПромКорм» зосереджений на втіленні спроектованої структури даних та логічних моделей у дієвий програмний комплекс. Головною метою програмної реалізації є створення гнучкого, швидкодіючого та інтуїтивно зрозумілого інтерфейсу, який забезпечує безперебійну взаємодію двох ключових категорій користувачів: зовнішніх контрагентів (агрогосподарств, фермерів) та внутрішніх менеджерів (адміністраторів підприємства). Програмний комплекс має розв'язувати задачі автоматизації обліку комбікормової номенклатури, спрощення процесу онлайн-замовлення, ведення аналітичного моніторингу та забезпечення прозорого зворотного зв'язку.

Вибір технологічного стека для реалізації клієнтської частини був обумовлений вимогами до кросплатформності, високої швидкості завантаження та автономності функціонування системи. Базовим інструментарієм виступає триада стандартних вебтехнологій: мова розмітки HTML5, каскадні таблиці стилів CSS3 та мова програмування JavaScript (ванільний JS без використання важких зовнішніх фреймворків) [9]. Такий підхід дозволив мінімізувати накладні витрати на збирання проєкту, гарантувати абсолютну сумісність із будь-якими сучасними браузерами та забезпечити високу швидкість обробки подій безпосередньо на стороні клієнта. Розмітка сторінок реалізована за принципами семантичного кодування, що підвищує структурованість коду, тоді як стилізація базується на сучасних модулях Flexbox та CSS Grid, забезпечуючи адаптивність інтерфейсу до екранів з різною роздільною здатністю [10].

Важливою архітектурною особливістю розробленого програмного забезпечення є застосування концепції SPA (Single Page Application – односторінковий вебзастосунок). Весь функціонал платформи, включаючи каталог товарів, кошик, авторизаційні форми, особистий кабінет клієнта та

розгалужену адмін-панель, інтегрований у межах одного фізичного файлу. Динамічне перемикання між функціональними розділами та вкладками здійснюється програмно за допомогою JavaScript-сценаріїв, які керують видимістю відповідних DOM-контейнерів та маніпулюють CSS-класами активності.

Варто чітко підкреслити, що вибір архітектури SPA у поєднанні із синхронним клієнтським сховищем на поточному етапі реалізовано виключно як високоефективну модель для локальної демонстрації розгалуженої бізнес-логіки, призначену для презентації проєкту на персональному комп'ютері розробника. Цей підхід дозволяє повноцінно продемонструвати взаємодію між модулями, симулювати роботу з базами даних та перевірити ергономіку інтерфейсу без розгортання складних серверних потужностей. Водночас спроектована логіка, модульна структура JavaScript-функцій та стандартизовані формати обміну даними (JSON, CSV) закладають надійний фундамент для безшовного перенесення цього прототипу в майбутньому на повномасштабний комерційний рівень, де локальні масиви будуть замінені на повноцінні серверні API та реляційні хмарні бази даних.

Програмна логіка застосунку розділена на кілька взаємопов'язаних функціональних блоків, кожен з яких відповідає за життєвий цикл певного процесу. Ключове місце посідає модуль ініціалізації сесії та керування станом, який під час завантаження сторінки перевіряє наявність даних у сховищі, за необхідності наповнює систему первинними демонстраційними записами та визначає роль поточного користувача. Наступні рівні реалізації включають:

1. Інтерактивний модуль каталогу та фільтрації: відповідає за динамічний рендеринг карток продукції та миттєве сортування асортименту комбікормів за призначенням (для птиці, свиней, ВРХ тощо) на основі масивів даних.

2. Транзакційний модуль кошика та замовлень: здійснює математичні розрахунки вартості, контролює відповідність обсягів замовлення наявним складським залишкам та реєструє нові заявки в журналі.

3. Аналітичний модуль адміністратора (Дашборд): програмно генерує графічні звіти та діаграми за допомогою технології маніпулювання векторною графікою SVG, забезпечуючи візуалізацію динаміки продажів та моніторинг критичних залишків кормів.

4. Комунікаційний модуль підтримки: реалізує логіку створення та обробки тікетів зворотного зв'язку між клієнтом та менеджером.

Важливим елементом підсистеми розмежування прав доступу та індивідуалізації взаємодії з клієнтами є модуль реєстрації нових контрагентів. Оскільки розробка орієнтована на забезпечення автономного та швидкого розгортання інформаційної платформи агропідприємства, вся логіка перевірки, обробки та персистенції даних нових облікових записів зосереджена на стороні клієнта і реалізована за допомогою мови програмування JavaScript. Це дозволяє миттєво реагувати на дії користувача, зменшувати час очікування та забезпечувати інтерактивність інтерфейсу без необхідності виконання мережових запитів до віддалених серверів.

Реалізація зазначеного алгоритму покладається на інтерактивні сценарії обробки подій, які перехоплюють надсилання екранної форми реєстрації. Послідовність виконання операцій під час створення нового облікового запису має чітко детермінований характер і складається з кількох послідовних етапів: зчитування інформації з полів введення, комплексна клієнтська валідація, перевірка унікальності ідентифікатора в існуючому реєстрі та безпосередній запис об'єкта у локальне сховище.

На першому етапі програмний скрипт звертається до об'єктної моделі документа (DOM) для отримання текстових значень, введених користувачем у відповідні поля екранної форми: логін (username), повне найменування або ПІБ контрагента (name), адреса електронної пошти (email), контактний номер телефону (phone) та пароль доступу (password). Перед початком аналізу всі зчитані рядки піддаються обов'язковій попередній обробці за допомогою вбудованого методу trim(). Це дозволяє автоматично видалити випадкові

початкові та кінцеві пробіли, що запобігає виникненню логічних помилок при подальшій автентифікації.

Другим етапом є виконання комплексної валідації даних, яка є критично важливою для збереження цілісності інформаційної системи агропідприємства. Програмна логіка по черзі аналізує кожне поле на відповідність встановленим критеріям:

Перевірка на повноту заповнення: скрипт контролює, щоб жодне з обов'язкових полів не залишалося порожнім. У разі виявлення порожнього рядка виконання алгоритму переривається, а система динамічно інформує користувача про необхідність введення коректних даних.

Валідація структури електронної пошти: перевірка поля email здійснюється шляхом зіставлення введеного тексту з регулярним виразом (RegExp). Це дозволяє на етапі введення відсікти записи, які не відповідають стандартному формату поштової адреси (відсутність символу "@", доменної частини тощо).

Контроль довжини та надійності пароля: програмний код аналізує довжину введеної послідовності символів для забезпечення базового рівня безпеки облікових записів фермерських господарств та приватних покупців.

Третій етап алгоритму відповідає за забезпечення унікальності первинних ключів у системі. Програма зчитує з localStorage поточний масив користувачів db\_users та десеріалізує його з текстового формату за допомогою методу JSON.parse(). Далі, використовуючи вбудовані методи перебору масивів (наприклад, some() або find()), JavaScript-сценарій порівнює введений користувачем логін із кожним існуючим значенням атрибута username в інформаційній базі. Якщо збіг виявлено, система блокує подальшу реєстрацію, генерує попереджувальне повідомлення про те, що дане ім'я користувача вже зайняте, та пропонує змінити ідентифікатор.

Четвертий етап виконується лише за умови успішного проходження всіх попередніх перевірок. Скрипт конструює новий реляційний об'єкт користувача, присвоюючи кожному ключу відповідне валідоване значення. На

цьому рівні бізнес-логіки програма автоматично встановлює для створюваного облікового запису атрибут `role` із значенням `"client"`. Таке жорстке програмне кодування ролі за замовчуванням гарантує, що жоден новий користувач не зможе несанкціоновано отримати доступ до функцій модерації, складського обліку чи аналітичних дашбордів, які закріплені виключно за адміністративним персоналом підприємства.

Після формування об'єкта він додається до локального масиву за допомогою методу `push()`. Для остаточного збереження оновленого реєстру у пам'яті браузера викликається метод `localStorage.setItem()`, де масив об'єктів знову трансформується у текстовий рядок через `JSON.stringify()`.

Завершується робота модуля реєстрації динамічною зміною стану інтерфейсу користувача без перезавантаження вебсторінки, що повністю відповідає архітектурній концепції `Single Page Application`. Програма очищує вміст полів форми, модифікує класи видимості HTML-контейнерів (приховує секцію реєстрації та відображає вікно авторизації) і виводить на екран текстове сповіщення про успішне створення кабінету, запрошуючи контрагента увійти до системи під своїми новими обліковими даними.

```
function actionRegister() {
  const username = document.getElementById('input-reg-username').value.trim().toLowerCase();
  const name = document.getElementById('input-reg-name').value.trim();
  const email = document.getElementById('input-reg-email').value.trim();
  const phone = document.getElementById('input-reg-phone').value.trim();
  const password = document.getElementById('input-reg-password').value;

  if(!username || !name || !email || !phone || !password) {
    alert("Всі поля обов'язкові для реєстрації!");
    return;
  }

  let users = JSON.parse(localStorage.getItem('db_users'));

  const exists = users.some(u => u.username === username);
  if(exists) {
    alert(`Користувач з логіном "${username}" вже зареєстрований! Оберіть інший логін.`);
    return;
  }

  const newUser = { username, name, email, phone, password, role: "client" };
  users.push(newUser);
  localStorage.setItem('db_users', JSON.stringify(users));

  currentUser = newUser;
  localStorage.setItem('session_user', JSON.stringify(currentUser));
  updateAuthHeader();

  alert("Кабінет успішно створено. Ласкаво просимо!");
  switchPage('profile');
}
```

Рисунок 3.1 – Функція для реєстрації

The screenshot shows the 'Реєстрація кабінету клієнта' (Client Cabinet Registration) page. The header includes the 'АгроПромКорм Платформа' logo and navigation links for 'Головна', 'Каталог', and 'Кошик (0 кг)'. A 'Вхід / Реєстрація' button is in the top right. The registration form includes the following fields:

- Бажаний Логін (унікальний)**: Input field with placeholder 'Наприклад, agro\_sterap'.
- ПІБ / Назва компанії**: Input field with placeholder 'ФІ 'Олімп' або Петренко І.'.
- Електронна адреса**: Input field with placeholder 'example@domain.com'.
- Контактний телефон**: Input field with placeholder '+38 (000) XXX-XX-XX'.
- Пароль**: Input field with placeholder 'Створіть надійний пароль'.

Below the fields is an orange 'Зареєструйтесь' button and a link 'Вже є акаунт? Увійти сюди'.

Рисунок 3.2 – Сторінка реєстрації

Наступним функціональним компонентом підсистеми управління доступом є модуль автентифікації користувачів. Його головне завдання полягає у перевірці правомірності доступу суб'єкта до персонального кабінету або адміністративної панелі на основі зіставлення введених ідентифікаційних даних із записами, які зберігаються в інформаційній структурі платформи.

Програмна логіка процедури авторизації активується під час перехоплення події відправки форми входу. Алгоритм автентифікації виконується у такій послідовності:

По-перше, здійснюється програмне зчитування рядкових значень із відповідних екранних елементів введення – логіна та пароля доступу. Аналогічно до модуля реєстрації, до отриманих текстових масивів застосовується метод `trim()` для очищення від випадкових початкових або кінцевих пробілів, що мінімізує виникнення помилок під час автентифікації користувачів.

По-друге, виконується базова валідація на наявність незаповнених полів. Якщо одне з полів є порожнім, скрипт блокує подальшу обробку та динамічно виводить в інтерфейс відповідне застереження.

По-третє, програма звертається до локального сховища `localStorage`, вилучає текстовий масив `db_users` та перетворює його на масив об'єктів

JavaScript. За допомогою методу `find()` алгоритм здійснює пошук об'єкта, у якого атрибут `username` повністю збігається із введеним логіном. У разі успішного знаходження запису виконується наступна логічна перевірка – порівняння значення ключа `password` знайденого об'єкта з паролем, щойно введеним у форму.

Якщо користувача з таким іменем не існує в системі або якщо зафіксовано невідповідність символічних послідовностей паролів, JavaScript-код генерує помилку автентифікації. Система інформує відвідувача про введення некоректних даних і залишає форму активною для повторних спроб.

У разі ідеального збігу обох параметрів процедура авторизації вважається успішно завершеною. Програма фіксує успішний вхід шляхом запису ідентифікатора та ролі користувача у поточну сесію застосунку (глобальну змінну стану або спеціальний ключ `currentUser` у локальному сховищі), що дозволяє підтримувати авторизований стан при взаємодії з іншими сторінками платформи.

Завершальний етап процедури авторизації забезпечує динамічну перебудову інтерфейсу Single Page Application:

Викликається керуюча функція оновлення відображення елементів (наприклад, `updateAuthHeader()`), яка приховує кнопки входу та реєстрації, натомість виводить назву облікового запису контрагента, кнопку переходу до профілю та кнопку виходу із системи.

Програма аналізує атрибут `role` авторизованого об'єкта. Якщо роль визначена як `"admin"`, інтерфейс розблоковує видимість керуючого сайдбару та елементів доступу до адміністративного пульта, аналітичних інструментів та журналів експорту даних. Якщо користувач має роль `"client"`, система перенаправляє його до каталогу продукції або персонального кабінету, обмежуючи доступ до службових модулів.

Екранна форма авторизації приховується шляхом маніпулювання класами видимості DOM-контейнерів.

```

function actionLogin() {
  const userIn = document.getElementById('input-login-username').value.trim();
  const passIn = document.getElementById('input-login-password').value;

  if(!userIn || !passIn) { alert("Будь ласка, заповніть всі поля форми."); return; }

  const users = JSON.parse(localStorage.getItem('db_users'));
  const found = users.find(u => u.username === userIn && u.password === passIn);

  if(!found) {
    alert("Невірний логін або пароль користувача!");
    return;
  }

  currentUser = found;
  localStorage.setItem('session_user', JSON.stringify(currentUser));
  updateAuthHeader();

  alert(`Успішний вхід. Вітаємо, ${currentUser.name}!`);

  if(currentUser.role === 'admin') {
    switchPage('admin');
    switchAdminTab('dash');
  } else {
    switchPage('profile');
  }
}

```

Рисунок 3.3 – Функція для авторизації

Рисунок 3.4 – Сторінка авторизації

Програмна реалізація презентаційного шару та інформаційного базису головної сторінки вебплатформи виконана у вигляді інтерактивного програмного модуля, що поєднує статичну семантичну розмітку HTML5, каскадні стилі CSS3 та динамічні алгоритми генерації контенту на базі JavaScript. Головна сторінка є стартовим вузлом навігації в архітектурі Single Page Application і спроектована таким чином, щоб забезпечити одночасне

вирішення маркетингових, інформаційних та операційних завдань підприємства.

Декларативна структура головної сторінки визначена в межах контейнера `<section>` з унікальним ідентифікатором `page-home`. На рівні архітектури SPA початковий стан цього блоку містить CSS-клас `active`, що вказує на його пріоритетне відображення під час завантаження коду в браузер. Внутрішній простір секції розділений на три взаємопов'язані структурні сегменти:

1. Презентаційний блок (`.hero`): Містить заголовок першого рівня, текстовий опис призначення платформи та інтерактивний елемент виклику дії (`button`), який через обробник подій активує функцію маршрутизації `switchPage('catalog')` для миттєвого перенаправлення користувача до повного асортименту продукції.

2. Блок корпоративної інформації (`.section-block`): Реалізований за допомогою двоколонкової сітки CSS Grid (`.about-grid`), де ліва частина містить текстове обґрунтування технологічних переваг комбікормового виробництва, а права – сітку статичних віджетів аналітичних та виробничих метрик підприємства (досвід на ринку, показники натуральності сировини, річні обсяги випуску та мінімальний рівень реклаमाцій).

3. Зона динамічної вітрини: Представлена заголовком другого рівня та порожнім цільовим DOM-контейнером `<div>` з ідентифікатором `popular-products-root`, який виступає точкою монтування для динамічно згенерованих об'єктів.

Алгоритм функціонування модуля динамічного рендерингу головної сторінки інкапсульований у функцію `renderHomePage()`. Логіка її роботи базується на концепції документо-орієнтованої вибірки даних і складається з наступних програмних кроків:

```
function renderHomePage() {  
    const products = JSON.parse(localStorage.getItem('db_products'));  
    const root = document.getElementById('popular-products-root');
```

```

root.innerHTML = "";

const popularItems = products.filter(p => p.popular);
popularItems.forEach(p => {
    root.innerHTML += generateProductCardHTML(p);
});
}

```

Під час виклику функції комп'ютерний сценарій звертається до енергонезалежної пам'яті за допомогою методу `localStorage.getItem()`, вилучаючи текстову базу даних номенклатури під ключем `db_products`. Для перетворення рядка формату JSON у масив ітерабельних об'єктів JavaScript застосовується метод `JSON.parse()`. Наступним кроком є очищення цільового контейнера від залишків попереднього рендерингу шляхом присвоєння властивості `root.innerHTML` порожнього рядка, що запобігає дублюванню елементів інтерфейсу при повторній навігації.

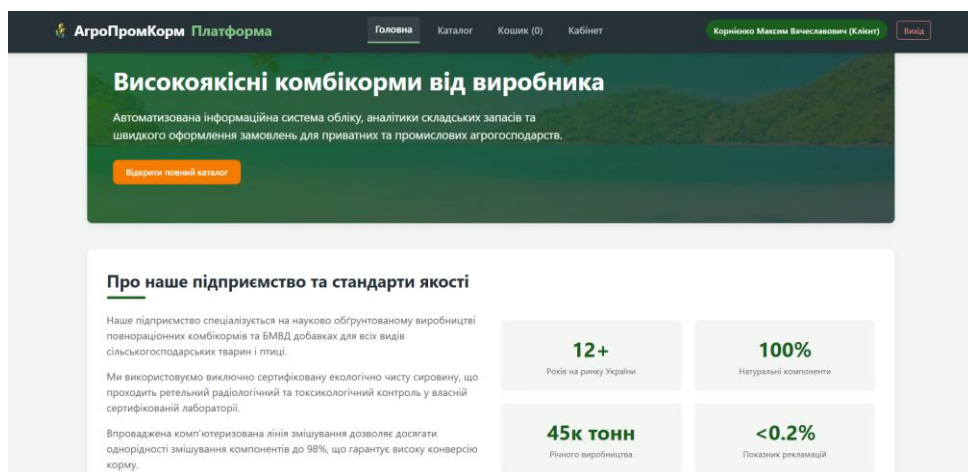


Рисунок 3.5 – Шапка та перші блоки головної сторінки

Для виділення актуальних сезонних пропозицій агропідприємства програма використовує високорівневий метод функціонального програмування масивів `.filter()`. Як предикат виступає стрілочна функція `p => p.popular`, яка аналізує булеве значення відповідного прапорця в кожному

об'єкті товару. Сформований відфільтрований масив об'єктів `popularItems` передається в циклічний обробник `.forEach()`, де для кожного елемента викликається допоміжна функція `generateProductCardHTML(p)`, а результат її роботи конкатенується із вмістом цільового DOM-вузла.

Допоміжний генератор `generateProductCardHTML(p)` реалізує паттерн проектування фабрики інтерфейсних компонентів на основі вхідних параметрів об'єкта продукту. Всередині функції розгорнуто умовне логічне розгалуження (`if / else if`), яке виконує семантичне картування текстового маркера категорії `p.img`. Залежно від зчитаного значення (наприклад, "poultry", "pig", "cow", "additive"), програма динамічно перевизначає змінні візуального представлення: шістнадцятковий код фонового кольору картки (`imgBgColor`) та людського читаний текстовий ярлик типу комбікорму (`bagLabel`).

Вихідне значення функції формується за допомогою механізму інтерполяції рядків (шаблонних літералів). У згенерований HTML-шаблон автоматично підставляються такі динамічні змінні об'єкта, як перелік цільових тегів призначення продукту (перетворених через метод `.map()`), найменування корму `p.title` та поточний кількісний показник доступної ваги на складі `p.stock`. Математичне форматування вартості реалізовано викликом вбудованого методу `p.price.toFixed(2)`, що приводить числове значення ціни до стандартного комерційного вигляду з двома знаками після коми.

Кожна згенерована картка містить функціональну кнопку додавання до кошика, у якій атрибут події `onclick` жорстко зв'язаний із викликом транзакційної функції системи `addProductToCart({p.id})`, де як унікальний параметр передається числовий ідентифікатор конкретної товарної позиції, забезпечуючи інтерактивну взаємодію користувача з платформою з перших секунд перебування на головній сторінці.

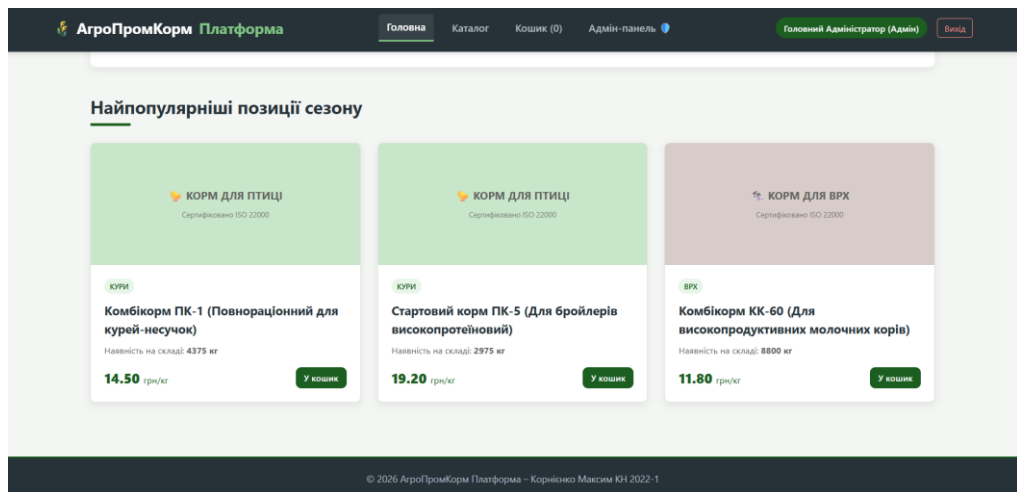


Рисунок 3.6 – Блок з актуальними сезонними пропозиціями на головній сторінці

Ключовим функціональним вузлом інформаційної системи, що забезпечує безпосередню взаємодію контрагента з товарним асортиментом підприємства, є модуль каталогу продукції. З точки зору користувацького інтерфейсу, сторінка каталогу побудована за допомогою CSS Grid-розмітки та розділена на дві основні робочі зони: бічну панель навігації з інструментами багатофакторної фільтрації та головний контейнер для динамічного виведення відсортованих карток комбікормів.

Програмна реалізація механізму пошуку та сортування базується на концепції реактивного оновлення контенту. Це означає, що система миттєво реагує на будь-які зміни критеріїв пошуку без необхідності перезавантаження сторінки або відправки додаткових запитів. Головним рушієм цього процесу є спеціалізована JavaScript-функція `applyFilters()`, яка автоматично викликається щоразу, коли користувач вводить текст у поле пошуку (подія `oninput`) або змінює стан будь-якого чекбокса-фільтра (подія `onchange`).

Алгоритм роботи функції `applyFilters()` складається з кількох послідовних етапів обробки даних:

1. Ініціалізація та зачитування критеріїв: На початку виконання скрипт звертається до локального сховища та завантажує актуальний масив усіх товарів `db_products`. Одночасно програма зчитує текстові значення з поля

пошуку `catalog-search`, негайно застосовуючи до нього метод `.toLowerCase()`. Це перетворення зводить введений текст до нижнього регістру, забезпечуючи нечутливість пошуку до великих чи малих літер.

2. Агрегація вибраних тегів: Для обробки фільтрів за призначенням (наприклад, "для курей", "для свиней") програма звертається до DOM-дерева за допомогою методу `document.querySelectorAll('.animal-filter:checked')`, що повертає колекцію всіх активних чекбоксів. Оскільки ця колекція не є стандартним масивом, її трансформують за допомогою `Array.from()`, а потім через метод `.map(cb => cb.value)` екстрагують значення активних фільтрів у єдиний плоский масив `checkedBoxes`.

3. Логічна фільтрація масиву: Центральною операцією алгоритму є застосування методу `.filter()` до загального масиву продукції. Предикатна функція виконує подвійну перевірку для кожного об'єкта товару `p`. Формується булева змінна `matchesSearch`, яка перевіряє входження пошукового запиту в назву товару за допомогою методу `p.title.toLowerCase().includes()`. Формується змінна `matchesTags`, що аналізує відповідність тегів. Якщо масив `checkedBoxes` порожній (фільтри не вибрано), змінна автоматично отримує значення `true`. В іншому випадку використовується метод `.some()`, який перевіряє, чи містить масив тегів поточного товару хоча б одне значення з вибраних користувачем. У підсумку товар залишається у відфільтрованому масиві лише за умови істинності обох умов (логічне "І": `matchesSearch && matchesTags`).

4. Динамічний рендеринг: Після завершення фільтрації програма очищує цільовий контейнер `full-catalog-root`. Перед початком генерації карток здійснюється перевірка на наявність результатів (`filtered.length === 0`). Якщо жоден товар не відповідає заданим критеріям, скрипт виводить форматоване інформаційне повідомлення про відсутність результатів, що є важливою практикою для покращення користувацького досвіду (UX).

5. Генерація об'єктів: Якщо відсортований масив не порожній, програма ітерує його за допомогою методу `.forEach()`, передаючи кожен об'єкт до вже

описаної фабричної функції `generateProductCardHTML(p)`. Згенерована HTML-розмітка додається до візуального контейнера.

Додатково в модулі реалізовано допоміжну функцію `clearFilters()`, яка забезпечує швидке скидання всіх налаштувань. Вона програмно очищує текстове поле пошуку, знімає позначки з усіх чекбоксів (перебираючи їх через `.forEach()`) та повторно викликає `applyFilters()` для відновлення повного асортименту каталогу.

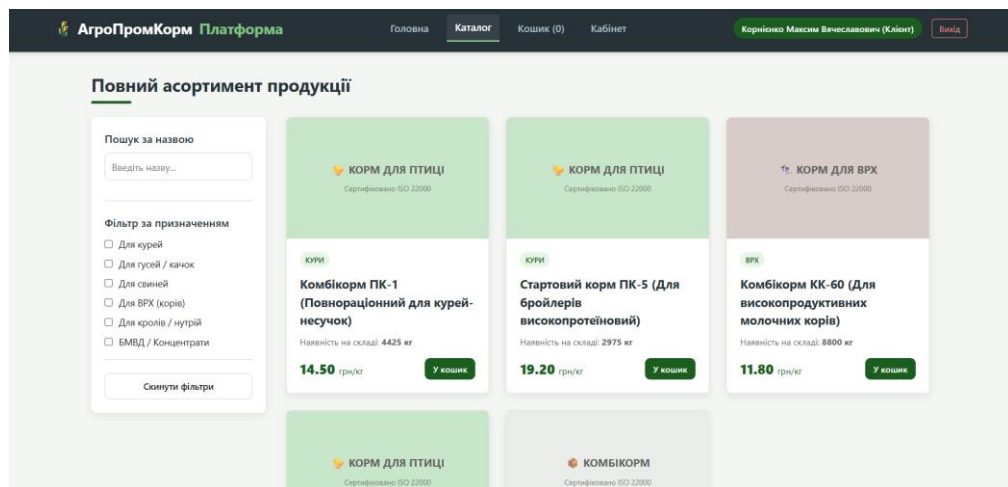


Рисунок 3.7 – Сторінка каталогу з бічною панеллю

Невіддільною складовою комерційного функціоналу платформи є модуль електронного кошика, який виконує роль транзитного буфера між вибором номенклатури в каталозі та фінальним оформленням замовлення. Оперативне зберігання вибраних товарних позицій протягом активної сесії користувача забезпечується глобальним масивом `userCart`. Такий підхід гарантує високу швидкість взаємодії, оскільки всі обчислення вартості та ваги відбуваються безпосередньо в оперативній пам'яті браузера без звернення до постійного сховища до моменту підтвердження транзакції.

Ініціація процесу покупки реалізується через функцію `addProductToCart(id)`. Алгоритм її роботи починається з пошуку замовленої позиції у локальній базі `db_products` за унікальним ідентифікатором. Програма здійснює первинну перевірку наявності товару на складі (`prod.stock <= 0`) і, у

разі його відсутності, негайно перериває виконання з виведенням попередження. Важливою особливістю бізнес-логіки підприємства, закладеною в цей метод, є продаж комбікормів стандартизованими мішками по 25 кілограмів. Якщо товар вже присутній у кошику, скрипт додає 25 кг до існуючого обсягу, попередньо перевіряючи, чи не перевищить нова кількість фактичний залишок на складі ( $\text{cartItem.qty} + 25 > \text{prod.stock}$ ). Якщо товар додається вперше, у масив `userCart` пушиться новий об'єкт із базовими атрибутами та початковою кількістю 25 кг.

Генерація візуального представлення кошика покладена на функцію `renderCartPage()`. Алгоритм використовує умовне розгалуження для визначення стану інтерфейсу: якщо масив `userCart` порожній, користувачу демонструється відповідний інформаційний блок із пропозицією перейти до каталогу, а таблиця розрахунків приховується. Якщо ж кошик містить товари, програма ітерує масив, динамічно формуючи HTML-рядки таблиці. Під час ітерації паралельно обчислюється загальна вартість кожної позиції (ціна за кілограм, помножена на вагу) та акумулюється підсумкова сума замовлення (`grandTotal`), яка згодом виводиться у фінансовому блоці.

Для гнучкого управління вмістом кошика реалізовані методи `changeCartQty(idx, offset)` та `removeCartItem(idx)`. Метод зміни кількості приймає індекс елемента в масиві та крок зміни (`offset`, який дорівнює 25 або -25). Програмний код містить захисні механізми: якщо нова кількість стає нульовою або від'ємною, товар автоматично видаляється з кошика викликом функції `removeCartItem`, а якщо запит перевищує складські запаси – генерується повідомлення про досягнення ліміту. Видалення товару реалізовано стандартним для JavaScript методом `splice()`, після чого відбувається примусове оновлення інтерфейсу та лічильника товарів.

Фінальним і найбільш комплексним етапом є підтвердження транзакції за допомогою функції `checkoutOrder()`. Алгоритм цієї функції розроблений з урахуванням вимог до цілісності даних і включає наступні кроки:

1. Перевірка авторизації: Система блокує оформлення замовлення для анонімних користувачів, вимагаючи наявності об'єкта `currentUser` у сесії, та перенаправляє неавторизованих клієнтів на сторінку входу.

2. Фінальна валідація залишків: Програма повторно звіряє кількість кожного товару в кошику з актуальними даними в `db_products`, щоб запобігти логічним колізіям (наприклад, якщо складські залишки були змінені адміністратором під час перебування клієнта на сайті).

3. Обробка даних та декрементация: У разі успішної перевірки, скрипт віднімає придбаний обсяг від поля `stock` відповідних товарів, формує текстовий перелік позицій (`itemsTextArr`) та розраховує підсумкову суму.

4. Реєстрація операції: Створюється новий об'єкт замовлення зі згенерованим ідентифікатором, поточною датою, прив'язкою до логіна користувача та базовим статусом "new". Цей об'єкт додається до масиву `db_orders`.

5. Синхронізація зі сховищем: Оновлені масиви продукції та замовлень серіалізуються і зберігаються в `localStorage`, після чого масив кошика `userCart` очищується, а користувач перенаправляється до особистого кабінету для відстеження статусу заявки.

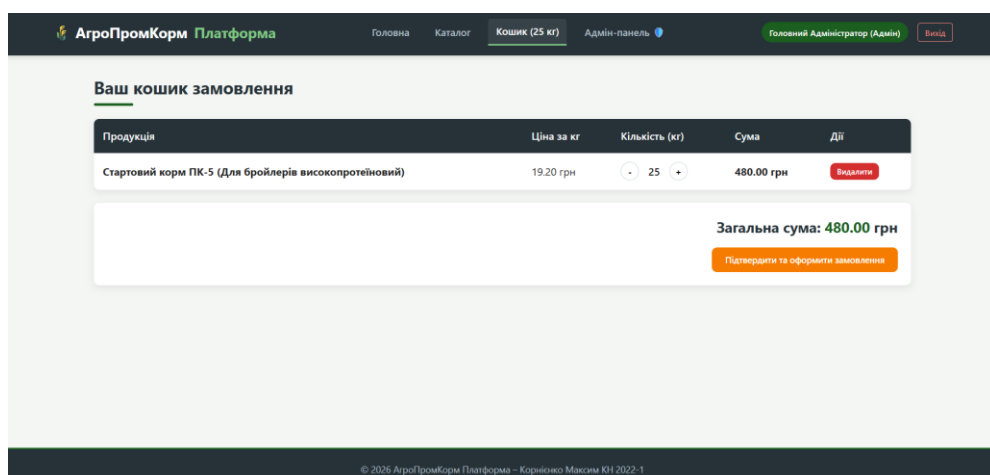


Рисунок 3.8 – Сторінка кошику із замовленням

Логічним центром персоналізованої взаємодії зареєстрованого контрагента з інформаційною вебплатформою є програмний модуль особистого кабінету. Цей компонент інтерфейсу спроектований як мультифункціональна панель, що об'єднує інструменти управління конфіденційними даними, моніторинг життєвого циклу замовлень та систему прямої комунікації з адміністративним персоналом агропідприємства.

Ініціалізація та наповнення цього модуля даними здійснюється за допомогою JavaScript-функції `renderProfilePage()`. Процес рендерингу розпочинається з перевірки наявності активної сесії: програма звертається до глобальної змінної `currentUser` і, у разі її валідності, програмно трансліює значення атрибутів об'єкта (ім'я, email, номер телефону) у відповідні поля введення HTML-форми. Важливою архітектурною деталлю є те, що поле ідентифікатора (логіна) блокується для редагування за допомогою DOM-властивості `disabled`, оскільки воно виступає первинним ключем для зв'язку з усіма іншими колекціями бази даних.

У межах цієї ж функції паралельно запускається алгоритм формування історії замовлень. Скрипт вилучає загальний масив транзакцій `db_orders` із локального сховища та застосовує до нього метод `.filter(o => o.username === currentUser.username)`, формуючи новий підмасив `myOrders`, що містить виключно покупки поточного клієнта. Під час ітеративного виведення рядків таблиці програма аналізує текстовий стан замовлення (`o.status`) і за допомогою серії умовних операторів динамічно призначає відповідні CSS-класи (`st-new`, `st-processing`, `st-shipped`, `st-delivered`), які візуально перетворюють системний статус на зрозумілий кольоровий бейдж (наприклад, "В обробці" або "Відвантажено").

Модифікація персональних даних реалізована через транзакційну функцію `saveProfileData()`. Після ініціації збереження алгоритм завантажує актуальний реєстр `db_users` та використовує метод `.findIndex()` для точного позиціонування поточного користувача в масиві. Значення з текстових полів перезаписують відповідні ключі знайденого об'єкта. Сценарій також містить

логіку безпечного оновлення пароля: перевіряється, чи не є поле нового пароля порожнім (`newPass.trim() !== ""`), і лише за істинності цієї умови атрибут `password` оновлюється. Після модифікації оновлений масив серіалізується назад у `localStorage`, а глобальна змінна сесії синхронізується з новими даними.

Крім того, користувачеві надано право на повне стирання свого цифрового сліду в системі через функцію `deleteMyProfile()`. Після проходження перевірки на підтвердження дії (`confirm()`), скрипт застосовує метод `.filter()` до колекції `db_users`, відкидаючи об'єкт поточного користувача, зберігає змінений масив і примусово викликає функцію деавторизації `actionLogout()`.

Окремим структурним підрозділом кабінету є модуль технічної підтримки та рекламацій. Формування нових звернень обробляється функцією `sendFeedback()`, яка зчитує категорію проблеми з випадаючого списку `<select>` та розгорнутий опис із текстового поля `<textarea>`. Скрипт генерує унікальний числовий ідентифікатор тикета за допомогою математичної функції `Math.floor(900 + Math.random() * 100)` та формує об'єкт зі статусом "open". Об'єкт додається до колекції `db_feedback`, після чого викликається допоміжна функція `renderUserFeedbackList()` для негайного оновлення інтерфейсу.

Функція `renderUserFeedbackList()` забезпечує відмальовування архіву діалогів. Вона здійснює вибірку тикетів авторизованого користувача, застосовує метод `.reverse()` для хронологічного сортування (від найновіших до найстаріших) і динамічно генерує HTML-блоки. Якщо об'єкт звернення містить непорожній атрибут `reply`, алгоритм рендерить блок із відповіддю від адміністрації; у протилежному випадку виводиться системне повідомлення про очікування на розгляд запиту.

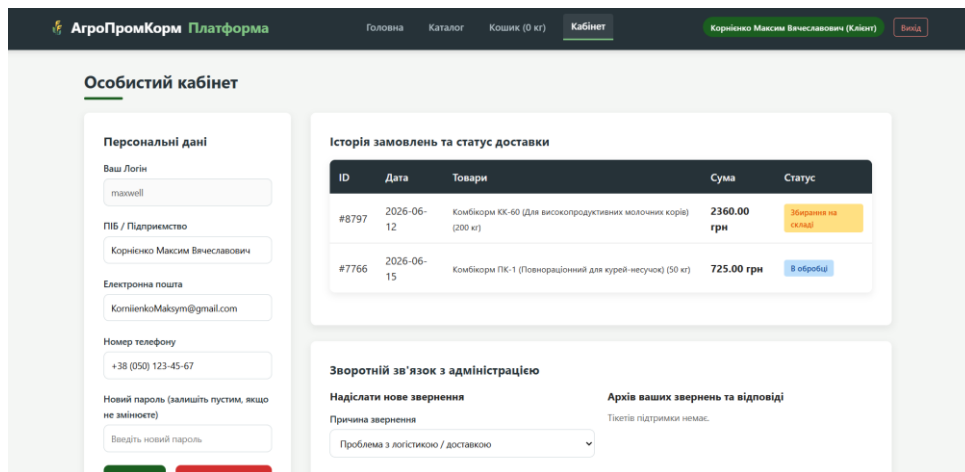


Рисунок 3.9 – Сторінка особистого кабінету з історією замовлень та блоком зворотного зв'язку

Головним інструментом управління бізнес-процесами, номенклатурою та клієнтською базою інформаційної вебплатформи є закритий модуль адміністративної панелі. На рівні архітектури Single Page Application доступ до цього компонента жорстко регламентується системою контролю ролей: відображення відповідного навігаційного посилання та завантаження DOM-контейнера можливе лише за умови, що глобальний об'єкт авторизованої сесії має атрибут `role` зі значенням "admin".

Програмний інтерфейс панелі керування реалізовано за патерном вкладок (tabs), що дозволяє компактно розмістити великий обсяг адміністративних інструментів на одному екрані без необхідності вертикального прокручування чи перезавантаження сторінки. Керування станом вкладок здійснюється через JavaScript-функцію `switchAdminTab(tabId)`, яка приймає ідентифікатор цільового розділу, циклічно видаляє CSS-клас `active` з усіх суміжних блоків та динамічно призначає його обраному контейнеру, паралельно викликаючи відповідну функцію рендерингу (наприклад, `renderAdminWarehouse()` або `renderAdminOrders()`).

Серцем адміністративного пульта є аналітичний дашборд, логіка якого інкапсульована у функції `renderAdminDashboard()`. Під час її виклику програма

синхронно зчитує всі чотири ключові масиви даних (`db_orders`, `db_products`, `db_users`, `db_feedback`) та розпочинає процес агрегації бізнес-метрик у реальному часі. Для обчислення загального фінансового доходу підприємства використовується метод згортання масиву `.reduce((sum, o) => sum + o.total, 0)`, який послідовно додає вартість кожного зареєстрованого замовлення. Кількісні показники (нові замовлення, критичні залишки сировини, кількість відкритих тікетів) розраховуються шляхом комбінації методів `.filter()` та властивості `.length`, після чого оновлені числові значення інтегруються у відповідні текстові віджети інтерфейсу.

Особливу увагу в архітектурі дашборду приділено модулю динамічної генерації векторної графіки (SVG), який відповідає за візуалізацію аналітичних звітів. З огляду на специфіку агробізнесу, замість статичної стовпчастої діаграми поточних складських залишків, було реалізовано складніший алгоритм візуалізації найпопулярніших видів кормів за обсягами продажів. Програма перебирає масив історії замовлень і застосовує регулярний вираз `/(.+?)\s\((\d+)\skg\)/` для лексичного аналізу текстового поля `items`. Цей підхід дозволяє екстрагувати назви товарів та придбану вагу (в кілограмах), знайти відповідний товар у базі `db_products` і додати цю вагу до загального лічильника конкретної категорії (тегу) кормів (наприклад, "кури", "свині", "врх"). Акумуляовані дані передаються до функції `drawPopularityChart()`, яка математично вираховує пропорції висоти стовпчиків відносно максимального значення та програмно створює елементи `<rect>` і `<text>` у просторі імен `http://www.w3.org/2000/svg`, формуючи наочний графік купівельного попиту.

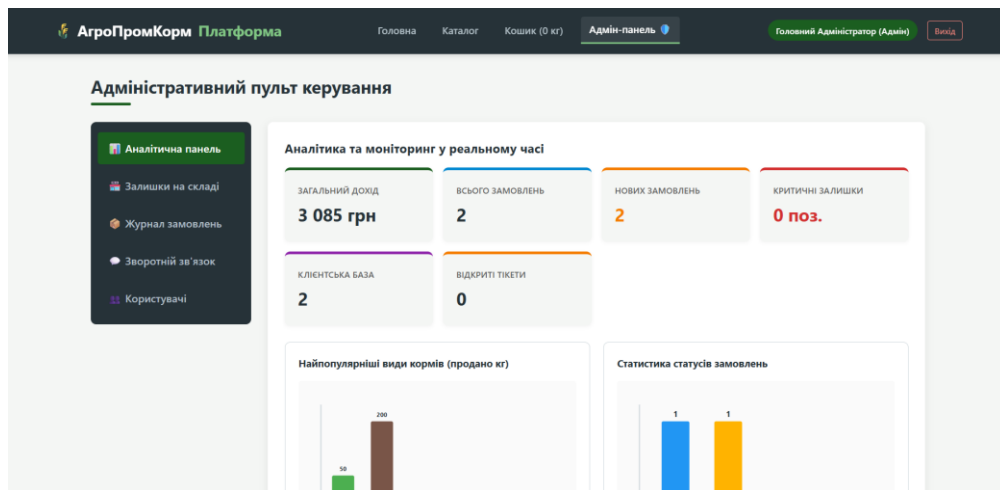


Рисунок 3.10 – Сторінка панелі адміністратора з відкритою аналітикою

Операційна взаємодія адміністратора з інформаційними потоками реалізована через такі функціональні підсистеми:

Диспетчеризація замовлень: Функція `renderAdminOrders()` генерує таблицю транзакцій, де для кожного запису передбачено інтерактивний елемент `<select>`. Зміна його значення ініціює функцію `changeOrderStatusAdmin()`, яка модифікує атрибут `status` відповідного об'єкта в локальному сховищі (від "В обробці" до "Доставлено"), що автоматично відображається в кабінеті клієнта.

Управління складом: Реалізовано через модуль додавання нової номенклатури `addNewProductAdmin()`. Скрипт збирає дані з полів (включаючи масив чекбоксів для тегів призначення), валідує їх на наявність числових значень та генерує новий об'єкт продукту з автоматично присвоєним унікальним ідентифікатором.

Обробка зворотного зв'язку: Функція `saveAdminReply()` дозволяє персоналу вводити текст відповіді на скарги чи пропозиції клієнтів. Алгоритм знаходить відповідний тикет у масиві `db_feedback`, заповнює атрибут `reply` та змінює стан звернення на "answered".

Завершальним і критично важливим архітектурним рішенням адміністративної панелі є центр автономного збереження та експорту даних, реалізований у функції `exportTableToExcelFile(type)`. Оскільки платформа

функціонує в середовищі браузера, виникла потреба у надійному механізмі передачі даних до зовнішніх бухгалтерських систем. Алгоритм формує структурований масив текстових рядків формату CSV (Comma-Separated Values), де значення відокремлюються крапкою з комою.

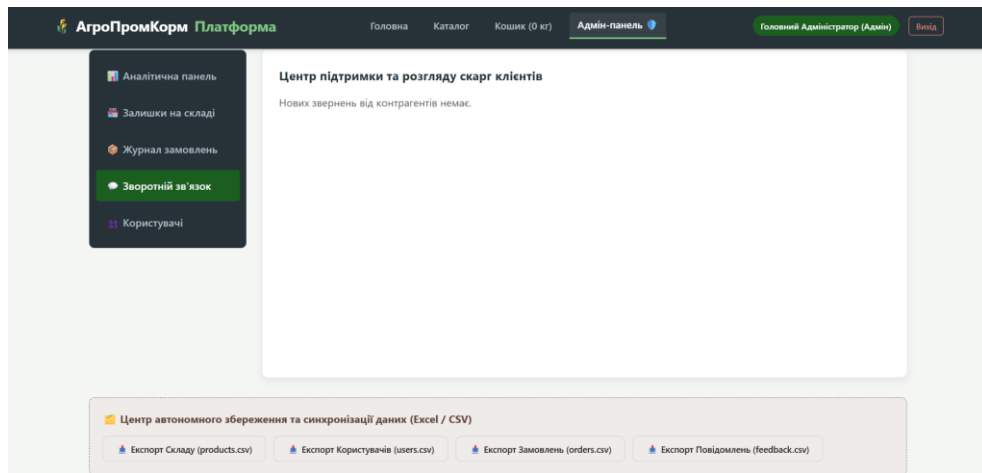


Рисунок 3.11 – Сторінка перегляду скарг клієнтів та блок експорту даних

Для забезпечення коректного відображення кирилических символів під час відкриття файлу в програмах Microsoft Excel, до початку згенерованого текстового масиву програмно додається специфічна мітка порядку байтів (BOM) – "\uFEFF". Фінальний текстовий блок інкапсулюється у бінарний об'єкт Blob із зазначенням MIME-типу text/csv;charset=utf-8. За допомогою методу URL.createObjectURL(blob) скрипт генерує тимчасове віртуальне посилання, програмно створює невидимий HTML-елемент <a>, імітує клік по ньому та ініціює збереження файлу на локальний диск комп'ютера адміністратора, після чого очищує пам'ять від тимчасових об'єктів.

### 3.3 Керівництво користувача

Клієнтський модуль вебплатформи спроектований з акцентом на інтуїтивну зрозумілість та мінімізацію кількості кроків, необхідних для оформлення заявки на постачання комбікормів. Взаємодія зовнішнього

користувача (фермерського господарства або приватного покупця) з платформою складається з кількох послідовних етапів.

Для отримання повноцінного доступу до можливостей оформлення замовлень та зворотного зв'язку користувачу необхідно створити персональний обліковий запис. Для цього на головній панелі навігації слід натиснути кнопку «Вхід / Реєстрація», після чого обрати опцію створення кабінету клієнта. У формі реєстрації обов'язковими для заповнення є поля унікального логіна, повного найменування (ПІБ або назва підприємства), контактної електронної пошти, номера телефону та надійного пароля. Якщо користувач вже має обліковий запис, йому достатньо ввести свій логін та пароль у формі авторизації для входу в систему.

Ознайомлення з номенклатурою підприємства здійснюється у розділі «Каталог». Інтерфейс цього розділу обладнаний бічною панеллю фільтрації, яка дозволяє оптимізувати процес пошуку необхідного комбікорму. Користувач може використовувати текстове поле для пошуку за точною назвою або застосовувати чекбокси для сортування продукції за призначенням (наприклад, для курей, свиней, ВРХ, водоплавних птахів або білкові добавки). Усі зміни фільтрів застосовуються миттєво, автоматично оновлюючи вітрину товарів.

Обравши потрібну позицію, клієнт натискає кнопку «У кошик» на картці товару. Варто враховувати специфіку фасування агропідприємства: одне натискання кнопки автоматично додає до кошика стандартизований мішок вагою 25 кг. Для перевірки та редагування замовлення користувач переходить до розділу «Кошик». У цьому модулі представлено таблицю розрахунків, де відображається ціна за кілограм, замовлена вага та загальна сума позиції. Користувач має можливість збільшити або зменшити кількість продукції за допомогою кнопок «+» та «-», а також повністю видалити товар із переліку. Система автоматично контролює наявність продукції та не дозволить замовити вагу, що перевищує поточні складські залишки. Після фінальної

перевірки необхідно натиснути кнопку «Підтвердити та оформити замовлення», після чого заявка передається диспетчеру.

Після авторизації в системі користувачу стає доступним розділ навігації «Кабінет». Він виконує функцію персонального інформаційного центру та складається з трьох ключових блоків:

1. Персональні дані: Дозволяє переглядати та редагувати свою контактну інформацію (телефон, email) або змінювати пароль доступу.

2. Історія замовлень: Надає доступ до таблиці всіх здійснених транзакцій. Найважливішим елементом тут є динамічний індикатор статусу (наприклад, «В обробці», «Збирання на складі», «Відвантажено», «Доставлено»), який дозволяє клієнту в режимі реального часу відстежувати етап виконання своєї заявки.

3. Зворотний зв'язок: Модуль для прямої комунікації з адміністрацією. Для створення нового звернення клієнт обирає категорію проблеми (наприклад, «Проблема з логістикою» або «Питання щодо якості»), детально описує ситуацію у текстовому полі та відправляє тикет. Відповіді від адміністративного персоналу публікуватимуться у цьому ж блоці в архіві звернень. Крім того, клієнт може подавати через цю форму комерційні пропозиції щодо довгострокової співпраці.

Адміністративний модуль вебплатформи вже є закритим віртуальним середовищем, призначеним для комплексного управління операційними процесами агропідприємства. Доступ до цього функціоналу суворо регламентований і надається виключно після успішної автентифікації під обліковим записом, що має системну роль адміністратора. Після авторизації в головному навігаційному меню автоматично розблоковується пункт «Адмін-панель», який слугує точкою входу до пульта керування.

Інтерфейс адміністратора побудований за принципом вкладок, що дозволяє швидко перемикатися між різними аспектами управління підприємством без перезавантаження сторінки. Робота з кожним підрозділом вимагає дотримання певного регламенту дій.

Аналітична панель (Дашборд) є стартовим екраном адміністратора і призначений для оперативного моніторингу ключових показників ефективності (КРІ) у режимі реального часу. У верхній частині панелі розташовано сітку інформаційних віджетів. Менеджер може миттєво оцінити загальний фінансовий дохід від реалізованої продукції, загальну кількість транзакцій, кількість нових (необроблених) замовлень, обсяг клієнтської бази та наявність відкритих тікетів від служби підтримки. Окрему увагу слід звертати на віджет «Критичні залишки» (виділений червоним кольором), який сигналізує про вичерпання запасів певних видів комбікормів.

Нижче розміщено блок візуальної аналітики. Графік «Найпопулярніші види кормів» дозволяє візуально оцінити попит на продукцію в розрізі цільового призначення (корми для курей, свиней, ВРХ тощо), що необхідно для планування виробничих потужностей. Графік «Статистика статусів замовлень» показує поточне навантаження на логістичний відділ.

Блок «Остання активність» забезпечує хронологічне відстеження останніх подій на платформі, таких як надходження нових замовлень або скарг від клієнтів.

Контроль залишків на складі призначений для синхронізації віртуальної вітрини з фактичним станом складських приміщень підприємства. Для розширення асортименту адміністратор повинен заповнити спеціальну форму. Необхідно вказати повну комерційну назву комбікорму (або БМВД), актуальну ціну за один кілограм у гривнях та наявну стартову вагу на складі.

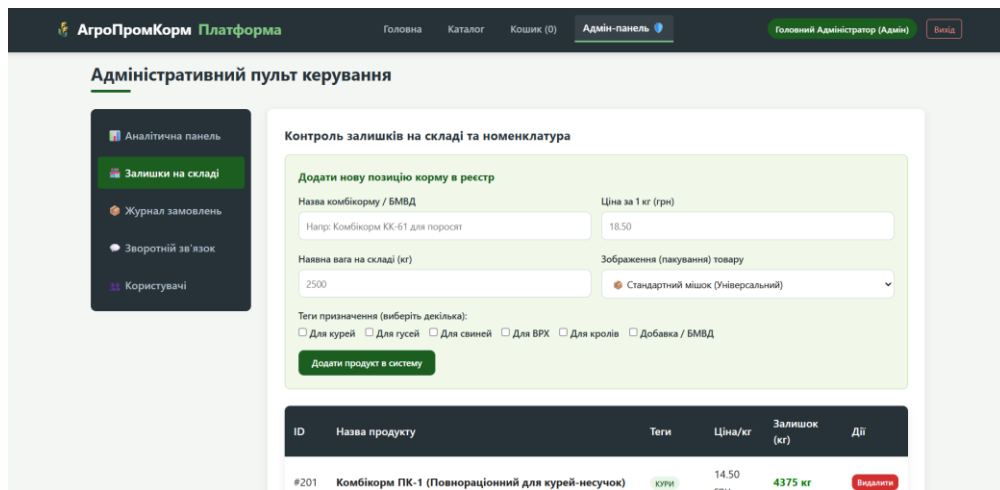


Рисунок 3.12 – Сторінка перегляду залишків на складі

З випадаючого списку обов'язково обирається тип пакування (наприклад, зелений мішок для птиці, жовтий для свиней або фіолетовий для добавок), що визначить кольорове оформлення картки товару в каталозі. Далі адміністратор повинен відмітити чекбоксами один або кілька тегів призначення (для курей, гусей, кролів тощо), щоб товар коректно відображався під час пошуку клієнтами. Після натискання кнопки «Додати продукт» нова позиція миттєво з'явиться в каталозі.

У нижній частині вкладки виводиться реєстр усіх товарів. Позиції, залишок яких становить менше 1000 кг, автоматично підсвічуються червоним кольором та маркуються попереджувальним знаком, що вимагає ініціації процесу до виробництва. Застарілі або зняті з виробництва позиції можна назавжди видалити за допомогою кнопки «Видалити».

У вкладці «Журнал замовлень» здійснюється управління логістичним циклом виконання заявок. Адміністратор бачить зведену таблицю всіх замовлень із зазначенням імені контрагента, дати, переліку придбаної продукції та загальної вартості. Ключовим інструментом диспетчера є випадаючий список «Зміна статусу» в кожному рядку таблиці. Під час фізичної обробки замовлення на підприємстві адміністратор зобов'язаний послідовно змінювати статус із «В обробці» на «Збирання складу», далі на «Відвантажено» і, після отримання підтвердження від водія, на «Доставлено».

Ці зміни синхронно відображаються в особистому кабінеті клієнта, забезпечуючи прозорість доставки.

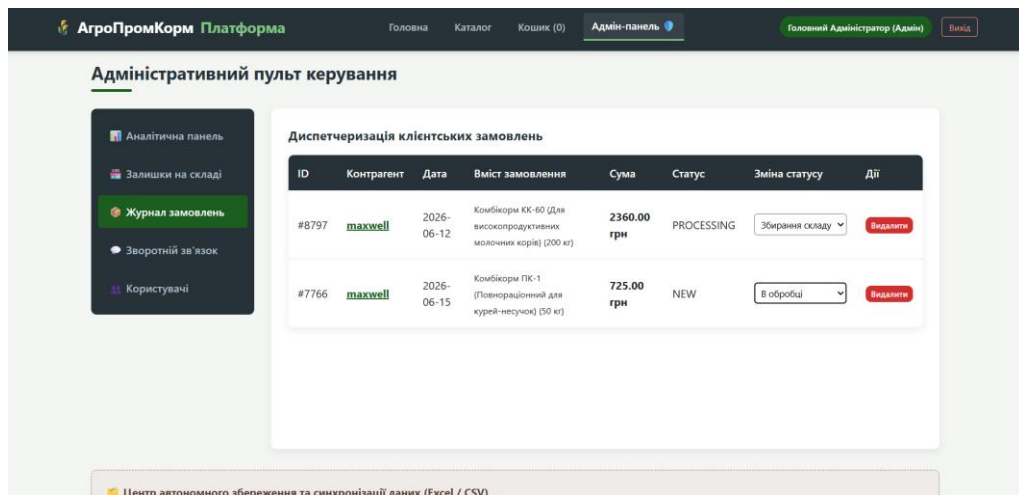


Рисунок 3.13 – Сторінка перегляду залишків на складі

Центр підтримки (Зворотний зв'язок) призначений для розв'язання конфліктних ситуацій, надання консультацій та розгляду комерційних пропозицій. Кожне нове звернення від клієнта відображається у вигляді окремого блоку із зазначенням дати, логіна автора, теми (наприклад, проблема з якістю чи доставкою) та повного тексту повідомлення. Для надання офіційної відповіді адміністратор вводить текст у відповідне текстове поле під повідомленням і натискає «Надіслати відповідь». Після цього тикет вважається закритим, а відповідь стає доступною клієнту. Тікети, що втратили актуальність, можуть бути видалені з бази.

Реєстр зареєстрованих користувачів надає доступ до клієнтської бази агропідприємства. Адміністратор має змогу переглядати список усіх контрагентів, їхні контактні телефони, електронні адреси та ролі в системі. У разі звернення клієнта (наприклад, телефоном) з проханням змінити неактуальні контактні дані, адміністратор використовує кнопку «Редагувати». Після цього відкривається прихована панель редагування, де можна оновити ПІБ, email або номер телефону клієнта та зберегти зміни в системі.

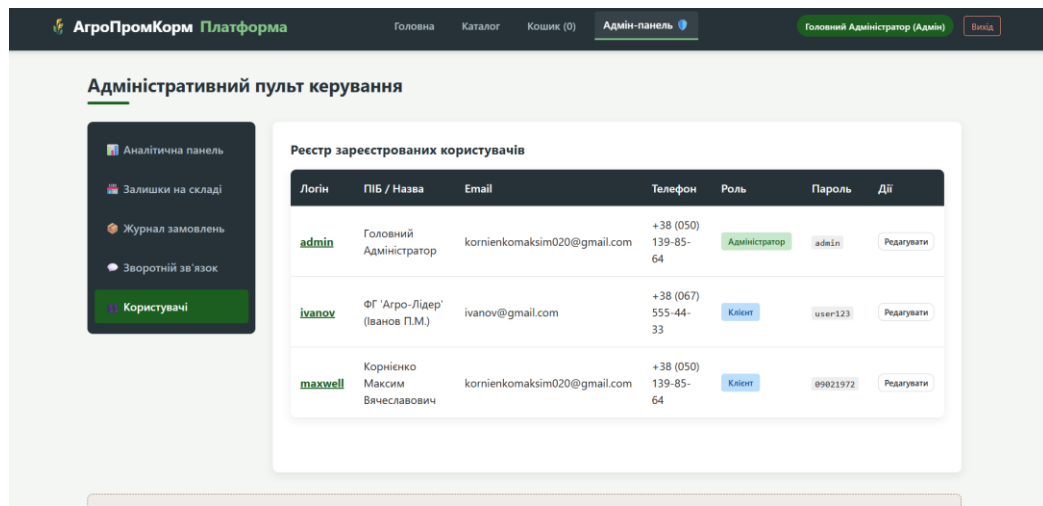


Рисунок 3.14 – Сторінка перегляду всіх зареєстрованих користувачів

Експорт даних зроблений для інтеграції вебплатформи з бухгалтерськими програмами або збереження резервних копій внизу адміністративної панелі передбачено блок експорту. Менеджер може натиснути відповідну кнопку для завантаження будь-якої колекції даних: Складу, Користувачів, Замовлень або Повідомлень.

Система автоматично згенерує та завантажить на комп'ютер файл у форматі CSV (наприклад, orders.csv), який повністю готовий до імпорту в табличні процесори (наприклад, MS Excel) або спеціалізовані облікові системи. Файли автоматично кодуються з підтримкою кирилиці, що унеможливорює спотворення текстової інформації.

### 3.4 Висновки до розділу 3

У третьому розділі кваліфікаційної роботи було здійснено повний цикл практичної реалізації розроблених раніше логічних та архітектурних моделей інформаційної вебплатформи агропромислового підприємства. Головним результатом цього етапу стало створення функціонального програмного комплексу, здатного автоматизувати процеси обліку номенклатури комбікормів та оптимізувати комерційну взаємодію з контрагентами.

На етапі проєктування підсистеми збереження даних було обґрунтовано вибір документо-орієнтованого підходу з використанням технології Web Storage API (localStorage). Таке рішення дозволило забезпечити високу швидкість обробки інформації та автономність роботи клієнтської частини без залучення сторонніх серверних потужностей. Було детально структуровано чотири базові реляційні колекції: реєстр користувачів, електронний каталог продукції, журнал транзакцій та архів тікетів зворотного зв'язку, що в комплексі повністю покривають операційні потреби підприємства.

Програмна реалізація інтерфейсу користувача та внутрішньої бізнес-логіки виконана мовою JavaScript із застосуванням концепції Single Page Application (SPA). У ході розробки було написано та інтегровано алгоритми клієнтської валідації для модулів реєстрації та авторизації, впроваджено механізми динамічної багатофакторної фільтрації асортименту в каталозі, а також реалізовано транзакційну математичну модель електронного кошика.

Особливу увагу було приділено створенню захищеної адміністративної панелі. Зважаючи на управлінські потреби агробізнесу, аналітичний дашборд було сфокусовано на генерації векторної візуалізації найпопулярніших видів кормів за обсягами продажів, що надає керівництву ефективний інструмент для планування виробничих навантажень.

Вирішено проблему ізольованості вебданих шляхом розробки модуля експорту. Впроваджений програмний механізм автоматично конвертує робочі масиви інформації у стандартизований формат CSV та ініціює їх завантаження у вигляді локальних файлів (з підтримкою кириличного кодування), що забезпечує безперешкодну інтеграцію вебплатформи з традиційними табличними процесорами та існуючими бухгалтерськими програмами підприємства.

Отже, створений програмний прототип підтверджує правильність обраних архітектурних рішень, задовольняє всі висунуті технічні вимоги та формує технологічний фундамент для подальшого масштабування системи у повноцінне комерційне середовище.

## РОЗДІЛ 4 ОХОРОНА ПРАЦІ

### 4.1 Організаційно-правові основи забезпечення безпеки праці

Охорона праці відіграє важливу роль у діяльності підприємств агропромислового комплексу, оскільки сфера АПК пов'язана з великою кількістю виробничих, технічних та логістичних процесів, під час яких працівники можуть піддаватися впливу небезпечних і шкідливих факторів. До таких факторів належать використання техніки та обладнання, переміщення вантажів, робота на складах, вплив шуму, пилу, несприятливих мікрокліматичних умов, а також значні фізичні та психоемоційні навантаження.

Основною метою охорони праці в агропромисловому комплексі є створення безпечних та комфортних умов праці, збереження життя і здоров'я працівників, попередження виробничого травматизму та професійних захворювань. Забезпечення належного рівня безпеки праці дозволяє не лише захистити працівників, але й підвищити ефективність роботи підприємства, зменшити кількість аварійних ситуацій та покращити організацію виробничих процесів.

Для агропромислових підприємств питання охорони праці має особливе значення через різноманітність видів діяльності. Працівники можуть виконувати роботи у виробничих приміщеннях, на складах, у логістичних центрах, працювати з транспортом, обладнанням та інформаційними системами. Саме тому система охорони праці в АПК повинна охоплювати всі напрямки діяльності підприємства та забезпечувати комплексний підхід до управління професійними ризиками [11].

Важливу роль у забезпеченні безпеки праці відіграє нормативно-правова база України. Основним документом, який гарантує право працівника на

безпечні умови праці, є Конституція України. У ній визначено, що кожен працівник має право на належні, безпечні та здорові умови праці.

Одним із головних нормативних документів у сфері охорони праці є Закон України «Про охорону праці». Цей закон визначає основні принципи державної політики у сфері охорони праці, права та обов'язки працівників і роботодавців, порядок організації безпечних умов праці та відповідальність за порушення вимог безпеки [11].

Важливе значення також має Кодекс законів про працю України, який регулює трудові відносини між працівником і роботодавцем, встановлює вимоги щодо режиму праці та відпочинку, робочого часу, умов праці та соціального захисту працівників [11].

Для підприємств агропромислового комплексу особливо актуальними є нормативні документи з:

- електробезпеки;
- пожежної безпеки;
- безпечної експлуатації обладнання;
- організації складських робіт;
- санітарно-гігієнічних умов праці.

Також важливими є державні санітарні норми та правила, які встановлюють вимоги до освітлення, вентиляції, мікроклімату, рівня шуму, організації робочих місць та роботи з комп'ютерною технікою.

У сучасних умовах агропромислові підприємства активно використовують інформаційні системи та вебплатформи для обліку продукції, управління логістикою та оформлення онлайн-замовлень. Це змінює характер праці працівників і додає нові фактори ризику, пов'язані з тривалою роботою за комп'ютером, інформаційним навантаженням та психоемоційним перенапруженням. Тому охорона праці в АПК сьогодні охоплює не лише виробничу безпеку, але й ергономічну організацію автоматизованих робочих місць.

Важливим елементом системи охорони праці є оцінка професійних ризиків. Вона дозволяє виявляти небезпечні фактори на робочих місцях, оцінювати рівень їх впливу на працівників та впроваджувати заходи щодо зниження ризиків. На основі результатів оцінювання підприємство може вдосконалювати умови праці, покращувати організацію виробничих процесів та підвищувати рівень безпеки персоналу.

Таким чином, охорона праці в агропромисловому комплексі є важливою складовою ефективної діяльності підприємства. Комплексне дотримання вимог нормативно-правових актів, організація безпечних умов праці та впровадження сучасних засобів управління ризиками дозволяють забезпечити захист працівників, зменшити рівень виробничого травматизму та підвищити ефективність роботи підприємств АПК.

#### 4.2 Характеристика об'єкта та виявлення потенційних небезпек

В розділі з охорони праці, враховуючи контекст дипломної роботи, буде розглянуто робоче місце працівника складу агропромислового підприємства, який може використовувати інформаційну вебплатформу, що розробляється.

Працівник складу агропромислового підприємства виконує роботи, пов'язані з прийманням, обліком, зберіганням, переміщенням та відвантаженням продукції. У сучасних умовах діяльність складських працівників тісно пов'язана з використанням інформаційних систем та вебплатформ для обліку продукції й оформлення онлайн-замовлень. Працівник використовує комп'ютерну техніку для ведення електронної документації, контролю залишків продукції, формування замовлень та координації логістичних процесів.

Таким чином, умови праці працівника складу поєднують офісну роботу за комп'ютером та фізичну діяльність безпосередньо на складських ділянках. Це створює комплекс небезпечних та шкідливих факторів різного характеру [12].

Робота на складі агропромислового підприємства часто виконується у великих приміщеннях із значною кількістю продукції, стелажів, складської техніки та транспортних засобів. Працівники можуть виконувати переміщення вантажів вручну або за допомогою навантажувачів, візків та іншого обладнання. Під час роботи необхідно постійно контролювати рух продукції, дотримуватись правил безпеки та взаємодіяти з іншими працівниками.

Одним із основних факторів є фізичне навантаження, пов'язане з переміщенням продукції, роботою з вантажами та тривалим перебуванням у русі. Піднімання та перенесення важких предметів може створювати значне навантаження на опорно-руховий апарат.

Можливі наслідки:

- біль у спині;
- перевтома;
- травми м'язів та суглобів;
- порушення роботи опорно-рухового апарату.

На складських ділянках існує ризик механічних пошкоджень під час роботи зі стелажми, інструментами, упаковкою або складською технікою. Працівник може отримати удари, порізи або травми внаслідок падіння предметів.

Можливі наслідки:

- забої;
- порізи;
- травми кінцівок;
- падіння працівника.

На складах агропромислових підприємств часто використовуються навантажувачі, рокли, електровізки та інша техніка для переміщення продукції.

Основні небезпеки:

- наїзд техніки;
- зіткнення;

- падіння вантажу;
- порушення правил експлуатації обладнання.

Складські приміщення можуть мати недостатнє опалення, вентиляцію або підвищений рівень вологості. У деяких випадках працівники працюють у холодних або запилених приміщеннях.

Можливі наслідки:

- швидка втома;
- простудні захворювання;
- погіршення самопочуття;
- подразнення органів дихання.

Підвищений рівень шуму – джерелами шуму можуть бути:

- складська техніка;
- транспортні засоби;
- переміщення вантажів;
- робота обладнання.

Постійний шум негативно впливає на нервову систему та концентрацію уваги.

На складах може зберігатися велика кількість упаковки, дерев'яних піддонів та легкозаймистих матеріалів. Порушення правил пожежної безпеки або несправність електрообладнання можуть спричинити займання.

Під час роботи з вебплатформою працівник виконує облік продукції, формує онлайн-замовлення та веде електронну документацію. Така діяльність супроводжується:

- тривалою роботою за комп'ютером;
- інформаційним навантаженням;
- психоемоційним перенапруженням;
- навантаженням на органи зору.

Особливо значне навантаження виникає при великій кількості замовлень або необхідності швидкого оновлення інформації.

Таблиця 4.1 – Детальна інформація про потенційні небезпеки

№	Небезпечний або шкідливий фактор	Джерело виникнення	Можливі наслідки
1	Фізичне навантаження	Переміщення та піднімання вантажів	Біль у спині, перевтома
2	Механічне травмування	Стелажі, вантажі, інструменти	Порізи, забої, травми
3	Падіння вантажів	Неправильне складування продукції	Травми працівника
4	Використання складської техніки	Навантажувачі, електровізки	Наїзд, зіткнення
5	Підвищений рівень шуму	Робота техніки та обладнання	Втома, дратівливість
6	Несприятливий мікроклімат	Холод, вологість, пил	Погіршення самопочуття
7	Пожежна безпека	Електрообладнання, упаковка	Пожежа, травмування
8	Тривала робота за комп'ютером	Використання вебплатформи	Втома очей, головний біль
9	Психоемоційне перенапруження	Велика кількість замовлень	Стрес, зниження уваги
10	Інформаційне перевантаження	Робота з електронною системою	Перевтома, помилки
11	Статичне навантаження	Робота за комп'ютером	Біль у шиї та спині
12	Електрична безпека	Комп'ютери та електрообладнання	Ураження струмом

#### 4.3 Дослідження ризику реалізації потенційних небезпек на об'єкті проектування та розробка заходів щодо їх попередження

Оцінка виробничих ризиків є одним із найважливіших елементів системи охорони праці, особливо на підприємствах агропромислового комплексу. Вона дозволяє своєчасно виявляти небезпечні та шкідливі фактори виробничого середовища, оцінювати рівень їх впливу на працівників та впроваджувати заходи щодо запобігання травмам, аваріям і професійним захворюванням.

Для підприємств агропромислового комплексу оцінка ризиків має особливе значення, оскільки діяльність у сфері АПК пов'язана з великою кількістю виробничих процесів, використанням техніки, складського обладнання, транспортних засобів та виконанням фізичних робіт. Працівники можуть одночасно працювати з вантажами, обладнанням, електротехнікою,

комп'ютерними системами та логістичними процесами, що створює комплекс небезпечних факторів різного характеру.

Основною метою оцінки ризиків є не лише виявлення небезпек, а й попередження виникнення небезпечних ситуацій. Саме завдяки оцінюванню ризиків підприємство може перейти від усунення наслідків нещасних випадків до їх профілактики. Це дозволяє значно підвищити рівень безпеки праці та зменшити ймовірність виробничого травматизму [13].

У процесі оцінки ризиків аналізуються всі умови праці працівників, обладнання, організація робочих місць, виробничі процеси та можливі джерела небезпеки. Для підприємств АПК це особливо важливо через різноманітність видів робіт та складні умови праці.

Оцінка ризиків дозволяє визначити, які саме фактори є найбільш небезпечними для працівників та які заходи необхідно впровадити для їх зниження. Наприклад, за результатами оцінювання можуть бути прийняті рішення щодо:

- покращення організації складських приміщень;
- встановлення додаткового освітлення;
- модернізації обладнання;
- забезпечення вентиляції;
- використання засобів індивідуального захисту;
- оптимізації логістичних процесів;
- удосконалення інформаційних систем;
- проведення додаткових інструктажів з охорони праці.

Особливу роль оцінка ризиків відіграє у сучасних умовах автоматизації підприємств АПК. Використання інформаційних вебплатформ для обліку продукції та онлайн-замовлень дозволяє підвищити ефективність роботи підприємства, але водночас створює нові фактори ризику, пов'язані з тривалою роботою за комп'ютером, високим інформаційним навантаженням та психоемоційним перенапруженням працівників. Саме тому оцінювання

ризиків повинно враховувати не лише виробничі, але й ергономічні та психофізіологічні аспекти праці.

Важливим результатом оцінки ризиків є підвищення ефективності системи управління охороною праці. Підприємство отримує можливість більш раціонально організувати робочі процеси, своєчасно виявляти небезпечні ситуації та контролювати стан безпеки праці на різних ділянках роботи.

Крім того, оцінка виробничих ризиків сприяє [13]:

- зниженню рівня виробничого травматизму;
- покращенню умов праці;
- зменшенню професійних захворювань;
- підвищенню продуктивності праці;

Проведемо оцінку ризику для небезпеки механічного травмування за допомогою методу «дерева відмов».

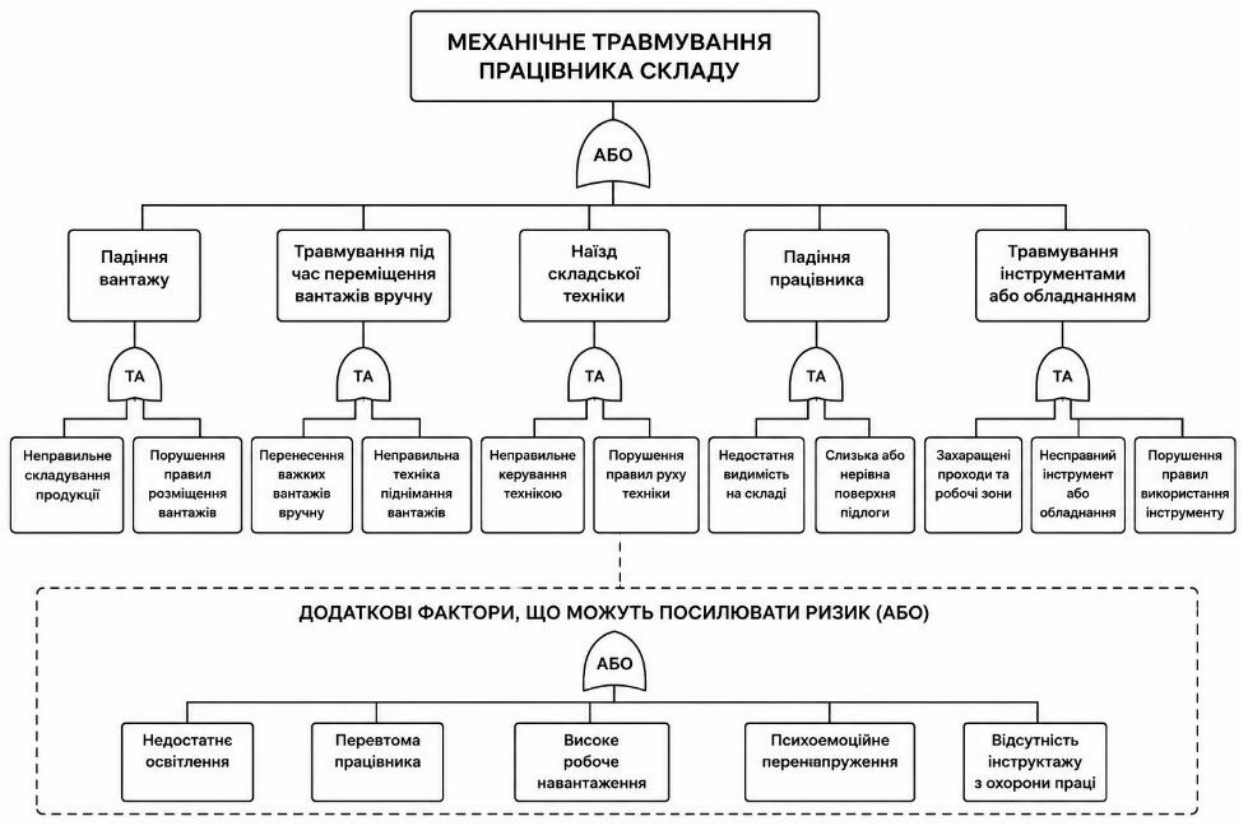


Рисунок 4.1 – Дерево відмов для небезпеки механічного травмування працівника складу

Для зниження ризику механічного травмування працівників складу агропромислового підприємства основна увага повинна приділятися правильній організації складських робіт, забезпеченню безпечних умов праці та контролю дотримання правил охорони праці [14].

Одним із найважливіших заходів є правильна організація складського простору. Проходи між стелажми повинні бути достатньо широкими та вільними від сторонніх предметів. Продукція має зберігатися відповідно до встановлених правил складування, а вантажі повинні бути надійно закріплені для запобігання їх падінню.

Для зменшення ризику травмування під час переміщення вантажів необхідно використовувати справну складську техніку та допоміжне обладнання. Працівники повинні дотримуватись правил експлуатації навантажувачів, рокл та електровізків, а також проходити відповідне навчання та інструктажі з безпеки праці.

Важливе значення має контроль технічного стану обладнання. Усі інструменти, стелажі, підйомні механізми та транспортні засоби повинні регулярно перевірятися на справність. Використання пошкодженого або несправного обладнання необхідно заборонити.

Для зниження ризику падіння працівників необхідно підтримувати чистоту та порядок у складських приміщеннях. Підлога повинна бути рівною та неслизькою, а місця з підвищеною небезпекою мають бути позначені попереджувальними знаками.

Особливу увагу слід приділяти освітленню складських приміщень. Недостатнє освітлення знижує концентрацію уваги працівників та збільшує ймовірність травмування. Робочі зони, проходи та місця навантаження продукції повинні мати достатній рівень освітлення.

Для зменшення фізичного навантаження доцільно механізувати процеси переміщення продукції та обмежувати ручне перенесення важких вантажів. Працівники повинні дотримуватись правил безпечного піднімання вантажів та

використовувати засоби індивідуального захисту, зокрема робоче взуття, рукавички та спецодяг.

#### 4.4 Висновки до розділу 4

Охорона праці відіграє важливу роль у діяльності підприємств агропромислового комплексу, оскільки сфера АПК пов'язана з великою кількістю виробничих, складських та логістичних процесів, під час яких працівники можуть піддаватися впливу небезпечних і шкідливих факторів. Забезпечення безпечних умов праці є необхідною умовою ефективної роботи підприємства, збереження здоров'я працівників та попередження виробничого травматизму.

У ході проведеного аналізу було розглянуто особливості умов праці працівника складу агропромислового підприємства, який використовує інформаційну вебплатформу для обліку продукції та онлайн-замовлень і одночасно виконує складські та логістичні операції. Встановлено, що діяльність працівника має комплексний характер та поєднує фізичні, технічні, інформаційні та психофізіологічні навантаження.

Особливу увагу було приділено оцінці виробничих ризиків та побудові дерева відмов для безпеки механічного травмування працівника складу. Проведений аналіз дозволив визначити основні причини виникнення небезпечних ситуацій та встановити фактори, які найбільше впливають на рівень безпеки праці.

На основі результатів дослідження було запропоновано комплекс заходів щодо покращення умов праці та зниження рівня професійних ризиків. Використання інформаційної вебплатформи для обліку продукції та онлайн-замовлень дозволяє підвищити ефективність роботи підприємства, зменшити кількість помилок, оптимізувати логістичні процеси та знизити вплив людського фактора. Водночас автоматизація процесів потребує належної

організації робочих місць та врахування психофізіологічних навантажень на працівників.

## ВИСНОВКИ

У ході виконання кваліфікаційної роботи було спроектовано та розроблено сучасну інформаційну вебплатформу для агропромислового підприємства, що спеціалізується на виробництві комбікормів та кормових добавок. На початковому етапі дослідження було проведено детальний системний аналіз предметної області, який дозволив виявити критичні проблеми традиційних методів управління компанією: інформаційну розсинхронізацію між складом і відділом збуту, високий ризик операційних помилок через «людський фактор» при ручному прийомі заявок, а також неможливість цілодобового обслуговування контрагентів. Результати цього аналізу та огляд існуючих конкурентних рішень дозволили сформувати чіткі вимоги до функціонала нової B2B/B2C цифрової екосистеми.

Розроблена система забезпечує повний та прозорий цикл взаємодії між виробником та клієнтами. Клієнтська частина платформи надає користувачам (фермерським господарствам та приватним особам) зручний доступ до інтерактивного каталогу з багатофакторною системою фільтрації за видами тварин, можливістю перевірки актуальних складських залишків у режимі реального часу та зручного формування замовлень через електронний кошик. Впроваджений особистий кабінет дозволяє відстежувати статуси виконання заявок та вести пряму комунікацію з адміністрацією через систему тікетів. Своєю чергою, закрита адміністративна панель пропонує персоналу потужний інструментарій для диспетчеризації замовлень, своєчасного контролю критичних залишків на складі, управління профілями користувачів та візуального аналізу купівельного попиту через динамічні графічні дашборди.

Програмну реалізацію вебплатформи виконано за архітектурою Single Page Application (SPA) з використанням надійного стека базових вебтехнологій (HTML5, CSS3, JavaScript). Для забезпечення швидкого розгортання, автономної роботи прототипу та локального збереження

інформації було імплементовано документо-орієнтований підхід на базі технології Web Storage API (localStorage) зі структурованими JSON-колекціями. Важливою технічною перевагою системи стала реалізація вбудованого модуля експорту масивів даних у формат CSV (із підтримкою кириличного кодування), що гарантує безперешкодну інтеграцію вебплатформи з існуючими бухгалтерськими процесорами та обліковими програмами підприємства.

Створений вебзастосунок пройшов комплексне тестування, яке підтвердило стабільність алгоритмів клієнтської валідації, коректність роботи механізму розмежування ролей (клієнт/адміністратор) та надійність математичних перерахунків складських запасів. Розроблена інформаційна вебплатформа є готовим інструментом для цифровізації збуту агропромислового підприємства, що оптимізує логістичні процеси, усуває проблему розрізненого товарного обліку та суттєво підвищує якість клієнтського сервісу. Використана модульна архітектура закладає міцний програмний фундамент для подальшого масштабування проєкту та його майбутньої міграції на повноцінну клієнт-серверну модель із використанням хмарних баз даних.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. FAO. The State of Food and Agriculture 2022: Leveraging Automation in Agriculture [Електронний ресурс]. – Rome : Food and Agriculture Organization of the United Nations, 2022. – 160 с. – Режим доступу: <https://www.fao.org/3/cb9479en/cb9479en.pdf> (дата звернення: 11.05.2026).
2. Рамський А. Ю. Цифровізація агропромислового комплексу: можливості та виклики // Бізнес Інформ. – 2022. – № 8. – С. 228–235.
3. Laudon K. C., Traver C. G. E-Commerce 2023: Business, Technology, Society. 19th ed. – New Jersey : Pearson Education, 2023. – 912 p.
4. W3C. Web Content Accessibility Guidelines (WCAG) 2.2 [Електронний ресурс]. – W3C Recommendation, 2023. – Режим доступу: <https://www.w3.org/TR/WCAG22/> (дата звернення: 11.05.2026).
5. OECD-FAO Agricultural Outlook 2022–2031 [Електронний ресурс]. – Paris : OECD Publishing, 2022. – Режим доступу: <https://doi.org/10.1787/f1b0b29c-en> (дата звернення: 11.05.2026).
6. Satzinger J. W., Jackson R. B., Burd S. D. Systems Analysis and Design in a Changing World. 8th ed. – Boston : Cengage Learning, 2022. – 636 p.
7. Web Hypertext Application Technology Working Group. HTML Living Standard : Web Storage [Електронний ресурс]. – WHATWG, 2024. – Режим доступу: <https://html.spec.whatwg.org/multipage/webstorage.html> (дата звернення: 20.05.2026).
8. MDN Web Docs. Working with JSON [Електронний ресурс]. – Mozilla Foundation, 2024. – Режим доступу: [https://developer.mozilla.org/en-US/docs/Learn\\_web\\_development/Core/Scripting/JSON](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Scripting/JSON) (дата звернення: 20.05.2026).
9. Haverbeke M. Eloquent JavaScript: A Modern Introduction to Programming. 4th ed. [Електронний ресурс]. – San Francisco : No Starch Press,

2024. – 472 p. – Режим доступу: <https://eloquentjavascript.net/> (дата звернення: 20.05.2026).

10. Murach J., Urban R. Murach's HTML and CSS. 5th ed. – Fresno : Mike Murach & Associates, 2022. – 608 p. – Режим доступу: <https://www.murach.com/shop/murach-s-html-and-css-5th-edition-detail> (дата звернення: 25.05.2026).

11. Аграріям саме час згадати вимоги законодавства з охорони праці у сільському господарстві – Режим доступу: <https://oppb.com.ua/news/agrariyam-same-chas-zgadaty-vymogy-zakonodavstva-z-ohorony-praci-u-silskomu-gospodarstvi> (дата звернення: 01.06.2026).

12. Інструкція з охорони праці під час робіт у складському господарстві. – Режим доступу: <https://esop.expertus.com.ua/formprint?fid=4646> (дата звернення: 01.06.2026).

13. Ризикоорієнтовані підходи в охороні праці. – Режим доступу: <https://oppb.com.ua/news/ryzykoooriyentovani-pidhody-v-ohoroni-pratsi> (дата звернення: 01.06.2026).

14. Дещо про безпеку у складських приміщеннях. – Режим доступу: <https://oppb.com.ua/news/deshho-pro-bezpeku-pratsi-u-skladskyyh-prymishhennyah> (дата звернення: 01.06.2026).